

FINAL REPORT

Online Hotel Reservation and Centralized Management System

Project Type: Build

Date: 01st May 2022

Abstract

Online reservation applications have become the primary requirement for any customer booking-based businesses when the customer base keeps expanding. A reservation system would be the first situation where a customer would interact with the business, so providing a pleasant initial-experience should be crucial. Data management systems are getting popular among different business types, as companies keep automating computer systems. When companies handle multiple businesses together, a centralized management would be the best choice to handle data.

Don Y.J Holdings (Pvt) Ltd is a reputed holding company, owning three hotels and two bungalows, which has no dedicated reservation systems. Third-party accounting software's are used to store company data, customer details and business reporting. The company doesn't have the freedom to provide additional personalized services based on the different customer types, as their reservations depend on online agencies. The company notices that as business rate increase, they are having difficulties in managing related data. Moreover, the company's accounting sector cannot efficiently create business reports as there is no central system to retrieve business data.

This project proposes an online customer reservation system and a centralized management system, linked together so that the company would only need to maintain one single application to manage majority of their automated tasks. The system will be based on web application technology, which contains various users including company higher authorities, employees and its customers. The reservation system would be dedicated web application system based on the hotels and bungalows owned by the company, so that every location can be managed simultaneously. Customers can browse through the company's daily reservation or customer offers. The application would be based on a company-dedicated database designed using the company requirements, and the application can be accessed remotely through the internet.

Table of Contents

1. Introduction & Literature Review.....	12
1.1 Introduction	12
1.2 Background and Motivation.....	12
1.3 Problem in brief.....	13
1.4 Aim & Objectives.....	14
1.4.1 Aim	14
1.4.2 Objectives	14
1.5 Scope	15
1.6 Deliverables.....	16
1.7 Literature Review	17
2. Analysis.....	19
3. Design.....	32
3.1 Design Techniques	32
3.2 System Overview	33
4. Product Implementation.....	40
5. Validation	47
6. Critical Review & Conclusion	96
6.1 Closing executive summary	147
6.2 Conclusion.....	148
References / Bibliography	149
Appendices.....	151

List of Figures

Figure 1: Automation Summary Diagram	19
Figure 2: Use Case Diagram	23
Figure 3: User Story - General Manager.....	28
Figure 4: User Story - Accountant	29
Figure 5: User Story - Receptionist.....	29
Figure 6: User Story - Guest	30
Figure 7: ER Diagram.....	32
Figure 8: Guest user login - Flowchart	33
Figure 9: Employee user login - Flowchart	34
Figure 10: Employee functionalities - Activity diagram.....	35
Figure 11: Room reservation - Flowchart.....	36
Figure 12: Room reservation - Activity diagram	37
Figure 13: User Roles	38
Figure 14: Context diagram	40
Figure 15: User interface design process	41
Figure 16: Wireframes	41
Figure 17: Wireframe - Guest Login.....	42
Figure 18: Wireframe - Guest Registration.....	42
Figure 19: Wireframe - Employee Login.....	43
Figure 20: Wireframe - Employee Registration.....	43
Figure 21: Wireframe - Hotel reservation system home page	44
Figure 22: Wireframe - Hotel reservation system gallery.....	44
Figure 23: Wireframe - Hotel reservation search page	45
Figure 24: Wireframe - Hotel reservation search page	45
Figure 25: Project workplan.....	51
Figure 26: Development folder structure	54
Figure 27: Interfaces - Homepage.....	55
Figure 28: Interfaces - Room rates.....	56
Figure 29: Interfaces - Contact Us	57
Figure 30: Interfaces - Reservation Cart	58
Figure 31: Interfaces - Guest Registration	59
Figure 32: Interfaces - Guest Login	60
Figure 33: Interfaces - Alternative Guest Login	60
Figure 34: Interfaces - Confirm Reservation	61
Figure 35: Interfaces - Reservation System Admin Login.....	62
Figure 36: Interfaces - New Room Type	62
Figure 37: Interfaces - View Available Room Types	63
Figure 38: Interfaces - Edit Available Room Type.....	63
Figure 39: Interfaces - Create New Room	64
Figure 40: Interfaces - View Available Rooms.....	64
Figure 41: Interfaces - Edit Available Rooms	65

Figure 42: Interfaces - View and Manage Reservations	65
Figure 43: Interfaces - View Reservation Record.....	66
Figure 44: Interfaces - Edit Reservation Record.....	67
Figure 45: Interfaces - View List of Administrators.....	67
Figure 46: Interfaces - Create New Administrator Account	68
Figure 47: Interfaces - Update Administrator Account.....	68
Figure 48: Interfaces - Room Booking Report Generation	69
Figure 49: Interfaces - Print Booking Report.....	69
Figure 50: Interfaces - Centralized Management System Login.....	70
Figure 51: Interfaces - Centralized Management Admin Dashboard	70
Figure 52: Interfaces - Create New Department	71
Figure 53: Interfaces - View and Manage Existing Departments	71
Figure 54: Interfaces - Add New Staff Member	72
Figure 55: Interfaces - View and Manage Existing Staff Records.....	72
Figure 56: Interfaces - Edit Staff Member Record.....	73
Figure 57: Interfaces - Create Employee Payroll.....	73
Figure 58: Interfaces - View and Manage Employee Payrolls	74
Figure 59: Interfaces - Generate Employee Salary Invoice	74
Figure 60: Interfaces - View and Manage Employee Leave Requests	75
Figure 61: Interfaces - View Employee Leave History	75
Figure 62: Interfaces - Management System Employee Dashboard.....	76
Figure 63: Interfaces - View and Print Salary Payment Invoices	76
Figure 64: Interfaces - Apply for Employee Leave.....	77
Figure 65: Interfaces - View Leave Status History.....	77
Figure 66: Database Structure - Online reservation system.....	78
Figure 67: Database Structure - Centralized management system.....	78
Figure 68: Guest login - user interface.....	79
Figure 69: Guest login interface - code snippet	80
Figure 70: Guest login verification - code snippet.....	81
Figure 71: Guest user registration - user interface	82
Figure 72: Guest user registration - code snippet #1.....	83
Figure 73: Guest user registration - code snippet #2.....	84
Figure 74: User registration validation - code snippet.....	85
Figure 75: Employee login - user interface.....	86
Figure 76: Employee login interface - code snippet	87
Figure 77: Employee login verification - code snippet.....	88
Figure 78: Employee registration - user interface.....	89
Figure 79: Sample employee registration - filled form.....	89
Figure 80: Employee registration - code snippet #1	90
Figure 81: Employee registration - code snippet #2	91
Figure 82: Employee registration validation - code snippet	92
Figure 83: Session implementation during user login - code snippet	93
Figure 84: Session implementation during user logout - code snippet	93
Figure 85: Development Tools Utilization - Sublime Text	94

Figure 86: Development Tools Utilization - XAMPP Server	94
Figure 87: Software validation and verification	95
Figure 88: Software Testing.....	98
Figure 89: Functional Testing.....	99
Figure 90: Functional testing procedure illustration	99
Figure 91: Integration test case 1 - Evidence #1	103
Figure 92: Integration test case 1 - Evidence #2	103
Figure 93: Integration test case 1.1 - Evidence	104
Figure 94: Integration test case 2 - Evidence #1	106
Figure 95: Integration test case 2 - Evidence #2	106
Figure 96: Integration test case 3 - Evidence #1	107
Figure 97: Integration test case 3 - Evidence #2	107
Figure 98: Integration test case 4 - Evidence #1	108
Figure 99: Integration test case 4 - Evidence #2	108
Figure 100: Integration test case 5 - Evidence #1	109
Figure 101: Integration test case 5 - Evidence #2	109
Figure 102: Integration test case 5.1 - Evidence #1	110
Figure 103: Integration test case 5.1 - Evidence #2	110
Figure 104: Integration test case 6 - Evidence #1	111
Figure 105: Integration test case 6 - Evidence #2	111
Figure 106: Integration test case 6.1 - Evidence #1	112
Figure 107: Integration test case 6.1 - Evidence #2	112
Figure 108: Integration test case 6.2 - Evidence #1	113
Figure 109: Integration test case 6.2 - Evidence #2	113
Figure 110: Integration test case 6.3 - Evidence #1	114
Figure 111: Integration test case 6.3 - Evidence #2	114
Figure 112: Integration test case 7 - Evidence #1	115
Figure 113: Integration test case 7 - Evidence #2	115
Figure 114: Integration test case 7.1 - Evidence #1	116
Figure 115: Integration test case 7.1 - Evidence #2	116
Figure 116: Integration test case 7.1 - Evidence #3	117
Figure 117: Integration test case 7.1 - Evidence #4	117
Figure 118: Integration test case 7.2 - Evidence #1	118
Figure 119: Integration test case 7.2 - Evidence #2	118
Figure 120: Integration test case 7.3 - Evidence #1	119
Figure 121: Integration test case 7.3 - Evidence #2	119
Figure 122: Integration test case 8 - Evidence #1	120
Figure 123: Integration test case 9 - Evidence #1	121
Figure 124: Integration test case 9 - Evidence #2	121
Figure 125: Integration test case 9.1 - Evidence #1	122
Figure 126: Integration test case 9.1 - Evidence #2	122
Figure 127: Integration test case 10 - Evidence #1	123
Figure 128: Integration test case 10 - Evidence #2	123
Figure 129: Integration test case 10.1 - Evidence #1	124

Figure 130: Integration test case 10.1 - Evidence #2	124
Figure 131: Integration test case 10.2 - Evidence #1	125
Figure 132: Integration test case 10.2 - Evidence #2	125
Figure 133: Integration test case 10.3 - Evidence #1	126
Figure 134: Integration test case 10.3 - Evidence #2	126
Figure 135: Integration test case 11 - Evidence #1	128
Figure 136: Integration test case 11 - Evidence #2	128
Figure 137: Integration test case 11 - Evidence #3	128
Figure 138: Integration test case 11.1 - Evidence #1	129
Figure 139: Integration test case 11.1 - Evidence #2	129
Figure 140: Integration test case 11.2 - Evidence #1	130
Figure 141: Integration test case 11.2 - Evidence #2	130
Figure 142: Integration test case 11.3 - Evidence #1	131
Figure 143: Integration test case 11.3 - Evidence #2	131
Figure 144: Integration test case 12 - Evidence #1	132
Figure 145: Integration test case 12 - Evidence #1	132
Figure 146: Integration test case 12.1 - Evidence #1	133
Figure 147: Integration test case 12.2 - Evidence #1	134
Figure 148: Integration test case 12.2 - Evidence #2	134
Figure 149: Integration test case 13 - Evidence #1	135
Figure 150: Integration test case 14 - Evidence #1	136
Figure 151: Integration test case 14 - Evidence #1	136
Figure 152: Integration test case 15 - Evidence #1	137
Figure 153: Integration test case 15.1 - Evidence #1	138
Figure 154: Integration test case 15.1 - Evidence #2	138
Figure 155: Non-Functional Testing.....	141
Figure 156: Non-functional testing procedure illustration.....	141
Figure 157: Client administrative staff feedback form	142
Figure 158: Employee feedback form result - question 01	143
Figure 159: Employee feedback form result - question 02	143
Figure 160: Employee feedback form result - question 03	144
Figure 161: Employee feedback form result - question 04	144
Figure 162: Employee feedback form result - question 05	145
Figure 163: Employee feedback form result - question 06	145
Figure 164: Employee feedback form result - question 07	146
Figure 165: Appendix A.1 - Client employee feedback form.....	151
Figure 166: Appendix B.1 - Guest Login	153
Figure 167: Appendix B.2 - Guest User Registration.....	154
Figure 168: Appendix B.3 – Search Room Availability	156
Figure 169: Appendix B.4 - Reservation System Admin Login.....	157
Figure 170: Appendix B.5 – View and Manage Room Types.....	158
Figure 171: Appendix B.6 - Add New Room Type.....	159
Figure 172: Appendix B.7 – View and Manage Rooms	160
Figure 173: Appendix B.8 - Add New Room	161

Figure 174: Appendix B.9 - View and Manage Reservations	162
Figure 175: Appendix B.10 - Booking Report Generation	163
Figure 176: Appendix B.11 - Management System Admin / Employee Login.....	164
Figure 177: Appendix B.12 - View and Manage Departments.....	165
Figure 178: Appendix B.13 - Add Department.....	166
Figure 179: Appendix B.14 - View and Manage Employees	167
Figure 180: Appendix B.15 - Add Employees.....	168
Figure 181: Appendix B.16 - View and Manage Staff Payroll.....	169
Figure 182: Appendix B.17 - Add Staff Salary.....	170
Figure 183: Appendix B.18 - View and Manage Staff Leave Requests	171
Figure 184: Appendix B.19 - Create New Staff Leave Request	172

List of Tables

Table 1: SWOT Analysis	15
Table 2: PEST Analysis	16
Table 3: User Case 01 - Guest user registration.....	24
Table 4: User Case 02 - Employee registration	25
Table 5: User Case 03 - Guest user login.....	26
Table 6: User Case 04 - Employee login	26
Table 7: User Case 04 - Room reservation	27
Table 8: Security mechanism - User type-based functionalities	39
Table 9: Development technologies.....	46
Table 10: Required technical resources	46
Table 11: Integration testing template	101
Table 12: Integration testing targets.....	101
Table 13: Integration test case 1	103
Table 14: Integration test case 1.1	104
Table 15: Integration test case 2	105
Table 16: Integration test case 3	107
Table 17: Integration test case 4	108
Table 18: Integration test case 5	109
Table 19: Integration test case 5.1	110
Table 20: Integration test case 6	111
Table 21: Integration test case 6.1	112
Table 22: Integration test case 6.2	113
Table 23: Integration test case 6.3	114
Table 24: Integration test case 7	115
Table 25: Integration test case 7.1	116
Table 26: Integration test case 7.2	118
Table 27: Integration test case 7.3	119
Table 28: Integration test case 8	120
Table 29: Integration test case 9	121
Table 30: Integration test case 9.1	122
Table 31: Integration test case 10	123
Table 32: Integration test case 10.1	124
Table 33: Integration test case 10.2	125
Table 34: Integration test case 10.3	126
Table 35: Integration test case 11	127
Table 36: Integration test case 11.1	129
Table 37: Integration test case 11.2	130
Table 38: Integration test case 11.3	131
Table 39: Integration test case 12	132
Table 40: Integration test case 12.1	133
Table 41: Integration test case 12.2	134

Table 42: Integration test case 13	135
Table 43: Integration test case 14	136
Table 44: Integration test case 15	137
Table 45: Integration test case 15.1	138
Table 46: Integration testing report.....	139
Table 47: Appendix B.1 - Guest Login.....	153
Table 48: Appendix B.2 - Guest User Registration	155
Table 49: Appendix B.3 - Search Room Availability	156
Table 50: Appendix B.4 - Reservation System Admin Login	157
Table 51: Appendix B.5 – View and Manage Room Types	158
Table 52: Appendix B.6 - Add New Room Type	159
Table 53: Appendix B.7 – View and Manage Rooms	160
Table 54: Appendix B.8 - Add New Room	161
Table 55: Appendix B.9 - View and Manage Reservations.....	162
Table 56: Appendix B.10 - Booking Report Generation	163
Table 57: Appendix B.11 - Management System Admin / Employee Login	164
Table 58: Appendix B.12 - View and Manage Departments	165
Table 59: Appendix B.13 - Add Department.....	166
Table 60: Appendix B.14 - View and Manage Employees.....	167
Table 61: Appendix B.15 - Add Employees	168
Table 62: Appendix B.16 - View and Manage Staff Payroll	169
Table 63: Appendix B.17 - Add Staff Salary.....	170
Table 64: Appendix B.18 - View and Manage Staff Leave Requests.....	171
Table 65: Appendix B.19 - Create New Staff Leave Request	172

Glossary of Terms

CSS: Cascading Style Sheets, which acts as presentation describer for HTML.

Company authority: Any governing body of a company, who are allowed to take administrative decisions such as owners, co-owners or managers.

Database: Collection of data saved in computers to store and access required information.

Deployment environment: The environment where a software is deployed and executed to perform its set functionalities.

HDD: Hard Disk Drive, which is a storage device used to store data locally.

Holding company: A company that owns multiple other companies and manages them together.

HTML: HyperText Markup Language is a markup language used to design web applications.

Management system: A software application developed for storing and managing data related to individuals or businesses, which will be managed by the company authorities.

OS: Operating System is a low-level software which runs a computer's basic functions.

PHP: Hypertext Preprocessor (recursive acronym).

RAM: Random Access Memory is a fundamental computer hardware used for computer tasks.

Software quality assurance: The process of testing and analysing software functionalities to ensure quality and user experience standards.

Web application: An application software designed to run on a web server through the internet. Web applications can be accessed remotely from anywhere, using an internet browser.

Web domain: A substitute which replaces the internet protocol addresses within web applications, to understand and differentiate them easier.

Web server: Web servers act as software and hardware that use protocols to respond to internet requests. Web servers also store web application data including its source code files.

1. Introduction & Literature Review

1.1 Introduction

As technology continues its rapid development among all industries around the world, hospitality is starting to notice the benefits. Hotel management software has evolved a long way in assisting hoteliers advance the way their business operates, and there is little doubt as to its transformative influence (Lacalle, 2021). This project is focused on solving a problem which most businesses ignore during its initialization, which is data collection. This project will be based the development of an online hotel reservation system linked with a centralized management system using web application technology. The project flow commences with advanced research on the use of management systems, data collection, designing, development and then quality assurance.

1.2 Background and Motivation

Don Y.J Holdings (Pvt) Ltd is a reputed holding company which consists of three tourist hotels and two bungalows. Currently, each hotel/bungalow has a separate system to store/access employee/client information. The company has decided to interlink the data from each hotel into a complete centralized management system where complex procedures can be carried out efficiently. The company is also currently handling business accounts using spreadsheets and third-party software packages. The company is also interested in a dedicated reservation system for users to reserve hotels and bungalows, which are currently done in the traditional method and through online travel agencies, where creative functionalities are limited.

Online reservation systems are also very convenient, considering how simple the procedure is compared to the traditional method of visiting or calling the hotel authorities. Third-party hotel agencies are also an effective method to list hotel reservations, but the freedom to add custom functionalities are limited, as hotels might have to change the procedures of reservations based on various elements like customer discounts, weather or special location-based seasons. Considering all hotel business aspects, automated reservation and management systems also have positive impacts to conduct more effective business. Once a management system in initialized, it would act as the major storage method, which is an efficient method to access and store data.

Don Y.J company currently doesn't have any automated management system or any dedicated online user reservation system. This would be the basic requirement to develop the business as it helps the company to gain customers by increasing awareness among internet users. It would be reasonable to build a centralized web application system to meet the multiple requirements for an effective hotel system. The system would consist of an online reservation web application system for customers to book hotel rooms efficiently, as well as a management system with interlinked company database system to manage the company's employee/customer information.

1.3 Problem in brief

Automated management systems are generally the basic and essential application to save and retrieve all user information including the company authorities, employees as well as the clients. This system will hold all the important data such as the user personal information and login credentials. For any company requiring a computerized system, a user management system would be the basic requirement. This would also act as the foundation in case the company decides to develop further complex automated functionalities.

Don Y.J company currently has separate data storage methodologies for storing user information, including personal and business information. The company is facing difficulties in accessing user information, as different types of data are stored in various locations instead of any centralized location. The company also doesn't have a dedicated online reservation system, but instead most reservations are obtained through third-party online agency websites. The company is unable to provide direct benefits to the customers as they have limited functionalities with online agencies.

The company's current requirements are to develop a dedicated online reservation system for customers to directly deal with the company business, as well as a centralized user management system to store collected company information for efficient accessibility.

1.4 Aim & Objectives

1.4.1 Aim

The aim of this project is to develop an online reservation and centralized management system with the use of web application technology.

1.4.2 Objectives

1. To research hotel reservation systems and centralizing management systems.
2. To study related technologies that can solve the problem.
3. To identify user and system requirements required for the architecture.
4. To improvise system user experience for better usage efficiency.
5. To design and develop a system for solving the problem.
6. To enable hotel administrators to add and update business information.
7. To enable hotel employees to view and manage daily operations.
8. To enable hotel guests to check availability of rooms for reservation.
9. To enable hotel guests to reserve rooms and view them efficiently.
10. To evaluate the implemented system.
11. To prepare the final documentation.

1.5 Scope

The scope of the solution of the current project is to develop a web application for Don Y.J Holdings, which includes an online hotel reservation system and centralized management system. The web application would be developed in consideration with the relevant tools and technologies which are being widely used in the software market currently. Similarly, proper project analysis techniques would also be used in the process. These analysis techniques would be a significant assistance to maintain the project progress flow in a systematic and efficient manner. The evaluation from these analysis techniques would encourage better decision making towards essential sections of the project.

SWOT analysis, which stands for Strengths, Weaknesses, Opportunities and Threats, is a technique to majorly figure out the strengths and weaknesses of any project. Project handlers can manually identify and have a clear SWOT analysis report to assist further decision making. The process itself of researching and identifying these points of the project will increase the situational awareness of the project handlers. The SWOT analysis of the current web application development project is identified and categorized below;

Table 1: SWOT Analysis

Strengths	Weaknesses
<ul style="list-style-type: none">• Powerful and straightforward approach to solve the current problems.• Proper project research and analysis.• Updated development technologies.• Efficient project advancement techniques.• Time and cost-efficient application.	<ul style="list-style-type: none">• Lack of employee technical knowledge could limit application efficiency.• Wrong system maintenance may cause application to malfunction.• Minor competition among other systems, as the current entrance is slightly late.
Opportunities	Threats
<ul style="list-style-type: none">• Ability to expand software features.• Additions to the software could automate entire operations for holdings company.• Possibility to expand business by merging system with already-popular systems.	<ul style="list-style-type: none">• Incorrect communication of requirements may cause errors in project.• Carelessness in system security policies could compromise system data.• Future government policies may cause certain system features to be reworked.

PEST analysis, which abbreviates for Political, Economic, Sociological and Technological, is a project evaluation technique to identify major opportunities in the project business. These opportunities might depend on expanding business assets or even spread the business for more customer reach. Businesses are able to attract various opportunities with proper utilization of software and internet technologies. The PEST analysis of the current web application development project is identified and categorized below;

Table 2: PEST Analysis

Political	Economic
<ul style="list-style-type: none"> • Changes in tax policies affecting costs. • New business regulations affecting the current businesses approaches. • Government party changes affecting views on businesses. 	<ul style="list-style-type: none"> • Company's economic stability affecting further business growth. • Currency exchange rates affecting the economic flow within the business. • Country's GDP (Gross Domestic Product) affecting customer reach.
Sociological	Technological
<ul style="list-style-type: none"> • Age profile changes resulting in supposed changes within the company's approach. • Generational changes affecting customer reach based on their solution approaches. • Educational profile changes affecting company employee recruitment. 	<ul style="list-style-type: none"> • Brand new technologies being created as simpler replacements for older ones. • Tools getting update changes could require additional knowledge by user.

1.6 Deliverables

The major deliverable of the current project would be the online hotel reservation and centralized management web application system, for Don Y.J holdings. This would be provided with the final documentation, including evaluation data from the analysis, development and finalization stages of the project. The project deliverable would also contain an application user guide, as an assistance for the company employees on the functionalities of the entire system.

1.7 Literature Review

As technology keeps rapidly developing around the world, businesses are beginning to discover the benefits of hospitality and simplicity for customers and users. Software applications have become an essential factor in all kinds of businesses. Software systems are constantly proving how time-saving and cost-efficient they have become, over proper utilization over task operations. Automating business procedures in companies are greatly advantages to both the company employees as well as its customers, majorly saving time on every-day tasks.

Considering a hotel business, it is essential that the company employees as well as its guests get a flawless and simple process in receiving their required service. Similarly, software solutions are the ultimate spot to provide flawless customer experience among businesses. There are countless advantages to a reliable hotel management software system. A proper hotel management application would massively reduce the time company employees spend on conducting administrative tasks manually. Whether its saving time on manual responsibilities or increasing room bookings, every component of a hospitality system should be dedicated towards the targets of improving efficiency and enhancing guest experience (Lacalle, 2021).

Considering similar systems, there are various project ideas that have even taken steps into advance stages of automated reservation systems. Similar projects have succeeded additions of linking creative functionalities in system to automate even advance tasks in order to save time and costs for business employees and its customers. As a reference for a basic hotel reservation system, (Priyadharshini & Joy, 2021) developed an integrated Hotel Management System that both administrators and customers can use. The admin will inform customers of the availability of rooms in various hotels, and customers will verify the availability of rooms in the desired hotel. Customers would be able to find out if a room in a particular hotel is available. They would be able to book available rooms in advance based on their needs to make their stay more comfortable. The aim of this project was to develop an automated hotel management system is to handle all aspects of the hotel's information and booking system. This application attempted to cover all operations that occur in residential hotels. They sought to demonstrate how data/information is processed in hotels in our project, automated Hotel Management System. The hotel management overview was achieved by splitting the project into different modules. Customers are offered various facilities such as check-in, check-out, and entry editing, as well as advance payments. Customer would have the option to cancel his or her reservation if he or she desires. Customer ID or customer name may be used to search for any customer or employee.

Similarly, (Akazue, 2016) conducted research on enhanced hotel management information system for multiple reservation booking, stating that the addition of a conditional statement to the check-out component of online hotel reservation model provides the ability that accommodates multiple bookings in one reservation form. Also, the additional conditional statement, which forms a loop between the check-in and check-out components, will help the customer reserve a choice room for the next visit. Time wasted in filling multiple reservation forms due to multiple check-in and check-out will be mitigated. (Akazue, 2016) also proposed an enhanced model for a hotel management system reservation module, where prospective customers can select multiple check-in and check-out actions on a single reservation. It accommodates customers with straight reservation check-in and check-out dates without any break, and it accommodates customers who have multiple check-in and check-out actions; it would do this on a single reservation form.

Moreover, (Delizo & Esguerra, 2013) developed and design an on-line hotel reservation and management system for the College of International Tourism and Hospitality Management of the Lyceum of the Philippines University, Batangas Campus. It presents user-friendly features that will familiarize CITHM students on the online hotel reservation system, evaluate it and highlight the benefits it can provide to the college and staff. In addition, it will purvey supplement material in their front desk operation course. The researchers used the System Development Life Cycle and Microsoft Web Developer 2008 as the programming language. The developed software served as a tool for the students of CITHM to familiarize them on how to operate an online hotel reservation system. The developed software was an effective aid for the instructors in teaching the basic operations of hotel reservation system to their students. It also provided online security to protect privacy and financial information of clients.

Considering security and data privacy, any data collected from the client for project development purposes should be collected with permission from company authorities. System architectures and databases will be designed based on the types of information the company has requested to be stored. The application will be designed to only collect information that the user has entered or agreed to share. User privacy of both the company as well as the users will be crucially considered in the project. The application would be subjected to be thoroughly tested for major vulnerabilities, to ensure that in case of system breaches, the system data will be secure. The quality assurance procedure of the project should also test user experiences, to make sure system functions work in the most efficient manner to save time for the user.

2. Analysis

Analysis for the current project would mainly require a problem statement. Analysis tools and techniques would be used to identify proper project procedures and assist decision-making.

Problem Definition:

Currently, Don Y.J company currently has separate data storage methodologies for storing user information, including personal and business information. The problem faced by the company currently is that there are difficulties in accessing user information, as different types of data are stored in various locations instead of any centralized location. The company also doesn't have a dedicated online reservation system, but instead most reservations are obtained through third-party online agency websites. The company has a problem providing direct benefits to the customers as they have limited functionalities with online agencies.

The hotel automated management systems are the basic and day-to-day maintained application to save and retrieve all user information including the company authorities, employees as well as the clients. An automated application would hold all the essential data such as the user personal information and login credentials. For any company requiring a computerized system, a user management system would be the basic requirement. Technically, the application would be the foundation, considering the company depends on the automated systems to take major decisions. Automating the processes of a holding company would directly affect various factors, such as;

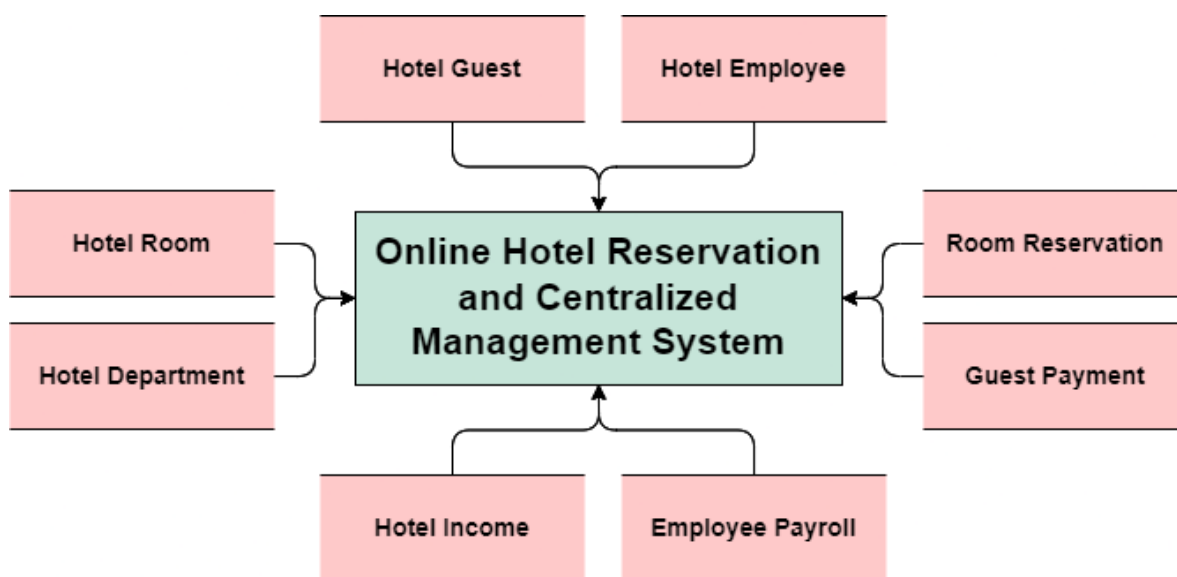


Figure 1: Automation Summary Diagram

Functional Requirements:

01. Application user roles and permissions.

- Don Y.J Holdings Administrator.
- Don Y.J Holdings Employee.
- Don Y.J Holdings Guest.

02. Administrator account.

- Create new administrator account.
- View list of available administrator accounts.
- Update administrator accounts records.
- Delete existing administrator accounts record.

03. Guest user registration.

- Create guest account.
- View guest account information.

04. User login.

- Administrator login
- Employee login.
- Guest login.

05. Room reservation.

- Search room availability based on dates and room types.
- Create new room reservation.
- View existing reservation details.
- Update reservation status.
- View list of room reservation history.

06. Generate reservation confirmation receipt.

07. Room type.

- Create new room type.
- View list of available room type records.
- Update existing room type records.
- Delete existing room type record.

08. Room.

- Create new room.
- View list of available room records.
- Update existing room records.
- Delete existing room record.

09. Generate reservation income report.**10. Hotel department.**

- Create new department.
- View list of existing department records.
- Update existing department records.
- Delete existing department record.

11. Hotel employee.

- Create new employee record.
- View list of existing employee records.
- Update existing employee records.
- Delete existing employee record.

12. Employee payroll.

- Create new employee payroll.
- View list of available employee payrolls.
- Delete existing employee payroll record.

13. Generate employee salary invoice.

14. Employee leave.

- Create new leave request.
- View history of leave requests.
- Update status of existing leave requests.

Non-Functional Requirements:

01. Hardware requirements.

- Minimum Pentium 4 or newer processor.
- Minimum 4GB of RAM.
- Minimum 500GB of HDD storage.
- Stable internet router connection.

02. Software requirements.

- Running Windows 8 / macOS Sierra 10.12 or newer system operating system.
- Internet browser, “Google Chrome” recommended.

03. Web application requirements.

- Web application domain name.
- Web server for application hosting.
- Database system for application data storage.

04. Application security and privacy.

- Protected with a login page with strict validations for data entering and viewing data.

05. Application data safety.

- Practice to export frequent database backups of the application storages.

Moreover, SWOT and PEST analysis techniques have been implemented for the current project in order to identify third-party factors related with the scope of the project. The SWOT analysis would illustrate the Strengths, Weaknesses, Opportunities and Threats supposed in the system. Meanwhile the PEST analysis would elaborate the Political, Economic, Sociological and Technological factors involved in the process of the project.

In order to identify system user functionalities, use case analysis methodology can be implemented. The user functionalities of the application can be illustrated using a use case diagram. The use case diagram would include the different types of users in the system, along with the different system functionalities in the application. The functionalities would be properly linked with the relevant users who have access to the specific functions.

The use case diagram for the current project is mentioned below;

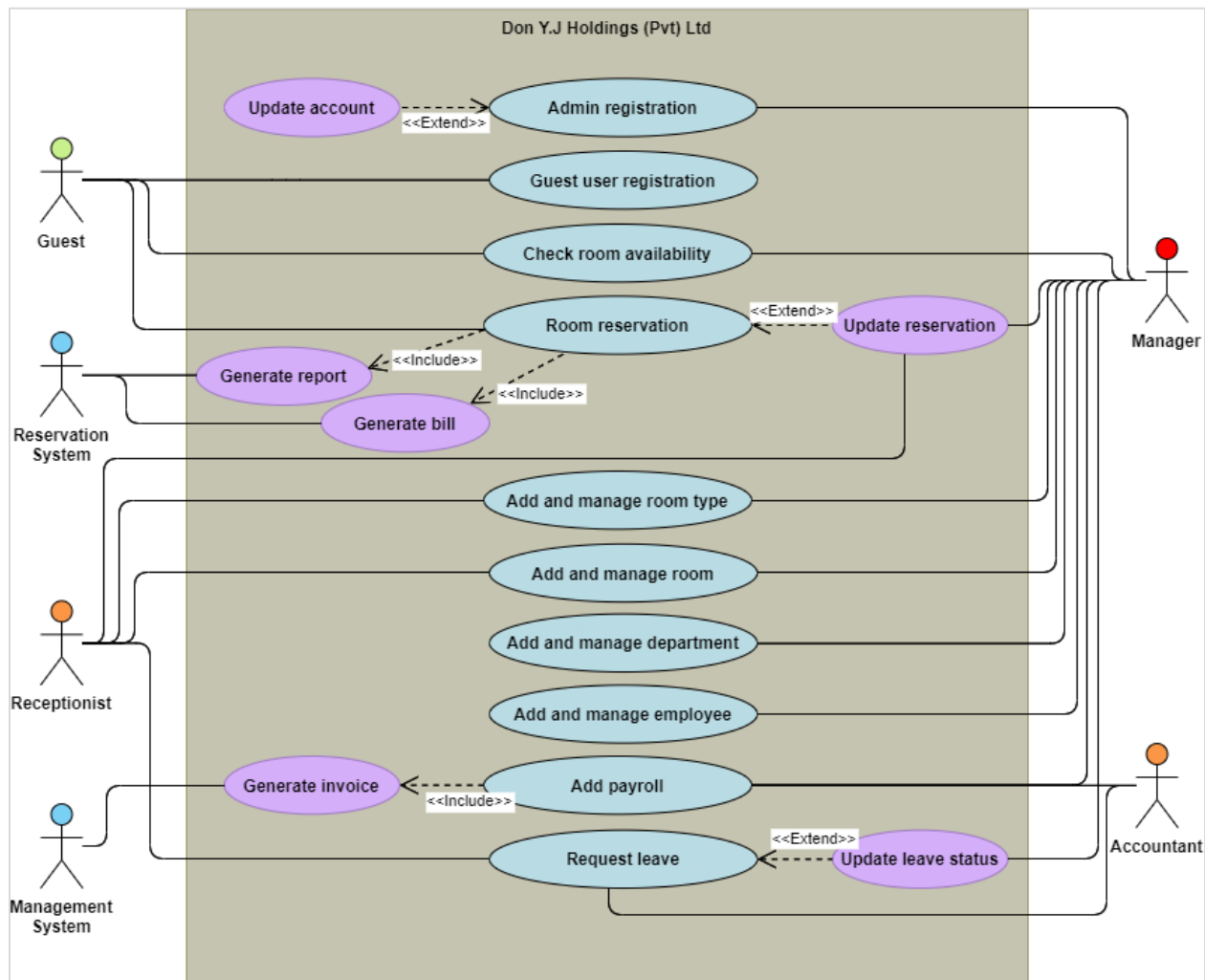


Figure 2: Use Case Diagram

Extended use case demonstrations could be used to analyse functional procedures in a step-by-step manner. The current project's essential functional use case demonstration is shown below.

The use case below demonstrates the system function of guest user registration.

Table 3: User Case 01 - Guest user registration

Use Case 01	Guest user registration
Actor	Guest
Functionality Flow	
<ol style="list-style-type: none"> 1. Guest is required to open web application using a web browser. 2. Guest is required to click “user registration” function. 3. Guest has to enter the following required personal information to register. <ul style="list-style-type: none"> - Full name - Date of birth - Address - Email address - Mobile number - NIC number - Username - Password 4. Guest should create a unique username with a strong password. 5. System would register user and provide a confirmation. 	
Alternate flow 1	If form finds mismatching record, an error would be displayed to edit it.
Alternate flow 2	If username is not unique, an error would be displayed to retry.
Alternate flow 3	If password is not strong enough, an error would be displayed to retry.
Pre-condition 1	Guest should not have an existing user account with similar records.
Post-condition 1	If registration successful, guest can login using username and password.

The use case below demonstrates the system function of employee registration.

Table 4: User Case 02 - Employee registration

Use Case 02	Employee registration
Actor	General Manager
Functionality Flow	
<ol style="list-style-type: none"> 1. Manager is required to open web application using a web browser. 2. Manager is required to login using username and password. 3. System redirects user to employee dashboard. 4. Manager has to enter the following required employee information to register. <ul style="list-style-type: none"> - Full name - Date of birth - Address - Gender - Email Address - Mobile number - Department - Registration date - Username - Password 5. System would register employee and provide a confirmation. 	
Alternate flow 1	If form finds mismatching record, an error would be displayed to edit it.
Alternate flow 2	If username is not unique, an error would be displayed to retry.
Alternate flow 3	If password is not strong enough, an error would be displayed to retry.
Pre-condition 1	Employee should not have an existing user account with similar records.
Post-condition 1	If registration successful, employee logs in using username and password.

The use case below demonstrates the system function of guest user login.

Table 5: User Case 03 - Guest user login

Use Case 03	Guest user login
Actor	Guest
Functionality Flow	
<ol style="list-style-type: none">1. Guest is required to open web application using a web browser.2. Guest is required to click “user login” function.3. Guest has to enter existing account username and password credentials.4. System would confirm successful login and redirect to homepage.	
Alternate flow 1	If credentials are mismatching, an error would be displayed to retry login.
Alternate flow 2	If user doesn’t have account, guest registration should be done.
Pre-condition 1	Guest should not have an existing user account.

The use case below demonstrates the system function of employee login.

Table 6: User Case 04 - Employee login

Use Case 04	Employee login
Actor	Employee
Functionality Flow	
<ol style="list-style-type: none">1. Employee is required to open web application using a web browser.2. Employee is required to access “employee login” function.3. Employee has to enter provided account username and password credentials.4. System would confirm successful login and redirect to employee dashboard.	
Alternate flow 1	If credentials are mismatching, an error would be displayed to retry login.
Pre-condition 1	Employee should have an existing employee account.

The use case below demonstrates the system function of room reservation.

Table 7: User Case 04 - Room reservation

Use Case 05	Room reservation
Actor	Guest
Functionality Flow	
<ol style="list-style-type: none"> 1. Guest is required to open web application using a web browser. 2. Guest is required to check room availability using below values. <ul style="list-style-type: none"> - Check-in date - Check-out date - Number of guests - Room type 3. Guest is request to select from the available rooms to create reservation cart. 4. Guest is required to click “cart” icon to update selection reservations. 5. Guest is required to click “continue reservation” button. 6. Guest is required to login to the application using existing guest account. 7. Guest would be required to confirm the reservation details. 8. Guest would get a reservation confirmation as a message to account. 	
Alternate flow 1	Rooms would not be shown is there is no availability within selection.
Alternate flow 2	If login credentials are mismatching, an error would be displayed to retry.
Alternate flow 3	If user doesn’t have account, guest registration should be done.
Alternate flow 4	If form finds mismatching record, an error would be displayed to edit it.

User Stories:

User stories are a great approach to analyse the feature of an application in the perspective of the end user. User stories depict the perspective of end users in a step-by-step manner. User storied would assist the evaluation on how valuable the system functionalities are for the user. Considering the current project, there would be four user types for the four user roles

The user story for the general manager user of the current project application is shown below.

General Manager																	
Function	Don't Loadings Web Application	User Login	Administrational Dashboard	Add Room Type	Manage Room Types	Add Room	Manage Rooms	Manage Reservations	Add New Admin	Manage Admins	Add Department	Manage Departments	Add Employee	Manage Employees	Add Staff Salary	Manage Payroll	Manage Leaves
User Story	Search and visit the web application using a web browser.	Enter username	Change dashboard view through navigation menu.	Enter room type	View available room types.	Enter room name	View available rooms	View available reservation details.	Enter name.	View available admins.	Enter department	View available departments	Enter full name	View available employees.	Enter basic salary	View employee payments.	View existing leave requests.
		Enter password		Enter description	Edit existing room types.	Select room type	Edit existing rooms.	Edit reservation status and details.	Enter username.	Edit admin records.		Edit existing departments.	Select department	Edit employee records.	Enter allowance.	Generate payroll invoice.	Approve leave request.
				Delete existing room type.	Enter description.	Delete existing rooms.	Delete reservation records.	Enter password.	Delete admin records.		Delete existing departments.	Select gender.	Delete employee records.	Verify total salary.	Delete payroll records.	Report leave request.	
					Enter number of guests.		Generate reservation income reports.	Select user type.			Enter email.						View employee leave history.
					Enter price.		Post reservation income reports.	Enter mobile number.			Enter mobile number.						
					Upload image.						Upload image.						
											Enter date of birth.						
											Enter date of joining.						
											Enter city.						
											Enter state.						
											Enter city.						
											Enter country.						
											Enter password.						
											Enter address.						

Figure 3: User Story - General Manager

The user story for the accountant user of the current project application is shown below.

Accountant							
Function	Don Y.J Holdings Web Application	User Login	Employee Dashboard	Manage Reservations	Add Staff Salary	Manage Payroll	Add Leave Request
User Story	Search and visit the web application using a web browser.	Enter username.	Change dashboard view through navigation menu.	View available reservation details.	Enter basic salary.	View employee payrolls.	Enter reason.
		Enter password.		Edit reservation details.	Enter allowance.	Generate payroll invoice.	Enter leave start date.
				Generate reservation income reports.	Verify total salary.	Delete payroll records.	Enter leave end date.
				Print reservation income reports.			Enter description.

Figure 4: User Story - Accountant

The user story for the receptionist user of the current project application is shown below.

Receptionist										
Function	Don Y.J Holdings Web Application	User Login	Employee Dashboard	Add Room Type	Manage Room Types	Add Room	Manage Rooms	Manage Reservations	Manage Payroll	Add Leave Request
User Story	Search and visit the web application using a web browser.	Enter username.	Change dashboard view through navigation menu.	Enter room type.	View available room types.	Enter room name.	View available rooms.	View available reservation details.	View existing payrolls.	Enter reason.
		Enter password.		Enter description.	Edit existing room type.	Select room type.	Edit existing rooms.	Edit reservation status and details.	Generate salary invoice.	Enter leave end date.
					Delete existing room type.	Enter description.	Delete existing rooms.	Delete reservation records.		Enter leave start date.
						Enter number of guests.		Generate reservation income reports.		Enter description.
						Enter price.		Print reservation income reports.		
						Upload image.				

Figure 5: User Story - Receptionist

The user story for the guest user of the current project application is shown below.

Guest							
Function	Don Y.J Holdings Web Application	Reservation System	Room Availability	Guest Registration	User Login	Reserve Room	Manage Reservations
User Story	Search and visit the web application using a web browser.	Home.	Enter check-in date.	Enter first name.	Enter username.	Select room.	View existing reservations
		Rooms.	Enter check-out date.	Select last name.	Enter password.	Add to cart.	View reservation bills.
		About Us.	Select number of guests.	Enter city.		Continue reservation.	
		Contact Us.	Select room type.	Enter address.		Confirm reservation.	
				Enter date of birth.			
				Enter email.			
				Enter profession.			
				Enter username.			
				Enter password.			
				Enter postal code.			

Figure 6: User Story - Guest

Analysis Outcome:

The system analysis phase of the current project was greatly advantage to understand further concept of the project idea. The system analysis procedures were implemented in a step-by-step procedure, in order to evaluate every aspect of the project idea in solving the company's problem. The analysis process typically comprises perceiving and assembling the required project fundamentals, to assist the process of moving towards the final project phases. Moreover, system analysis was considered as an essential decision-making aspect of the project. It is certain that this system analysis would be advantageous for the final phases of the product in order to increase the project efficiency and effectiveness.

The analysis process was also advantageous as it was split into multiple stages, mainly two types of data collection and data evaluation. Initially, the data related to the project was collected from various sources, mainly from the information provided by the company regarding the problem being faced and the required solution. This information was then analysed and categorized into system requirements, which would be the ultimate objectives of the current project deliverables. Eventually, extended researches were also conducted on the project area to get better understanding of the specific software field. This majorly includes analysing and evaluating similar project applications that have been already created and reported.

In summary, the following points have been studied from the conducted analysis;

- The system analysis methodologies are properly being identified and utilized.
- Project area knowledge has been gained, in assistance in formerly developed systems.
- The current project flow is moving in consideration with the initial objectives.
- The project objectives have been perfectly identified in order to solve context problems.
- The project deliverables are properly structured to provide appropriate client solutions.
- The system development methodologies are properly identified for an effective system.

3. Design

3.1 Design Techniques

The overall system planning of the project application could be illustrated using an Entity-Relationship Diagram (ERD). An ERD illustrates the relationships between entities and the users in a system, along with their interactions towards the application.

The ERD for this project would illustrate all key components and functions in the system, along with how they interact with each other. The ER diagram for this project is shown below;

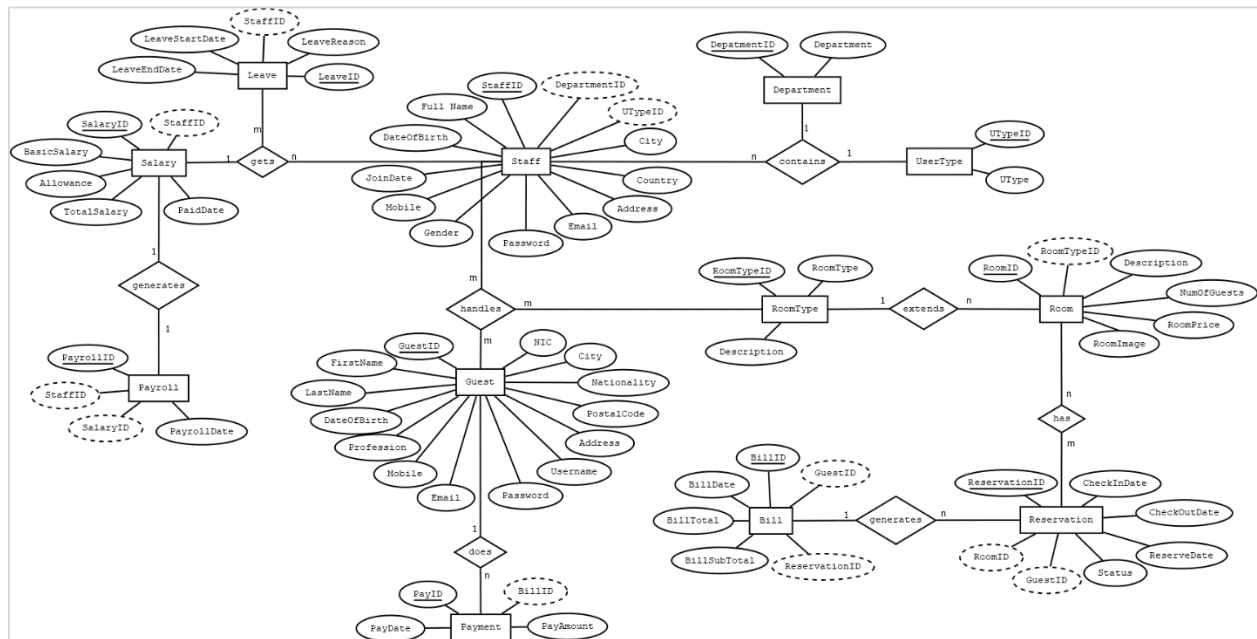


Figure 7: ER Diagram

User login could be considered as a basic functionality of the application. Both the company employees and the guests are required to login to the application using a username and password, to access essential system functions. If the guest doesn't already have a system account, they will be prompted to create one after entering guest information.

This process of guest user login can be illustrated using a flowchart as follows;

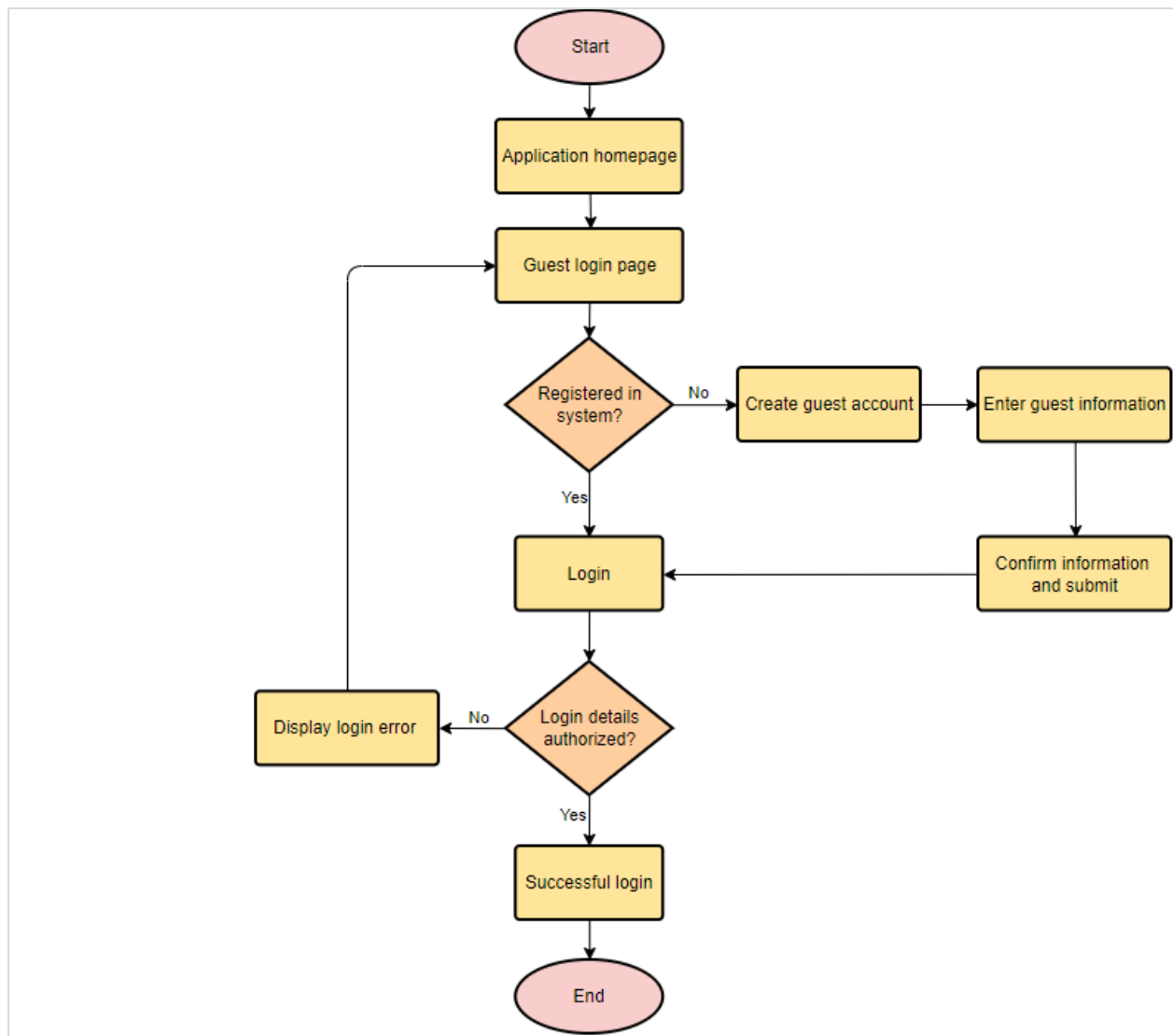


Figure 8: Guest user login - Flowchart

Company employees would receive their login credentials from the general manager, who would register and approve them in the system. Employees can then login using their unique username and password, in the employee login page of the system.

This process of employee user login can be illustrated using a flowchart as follows;

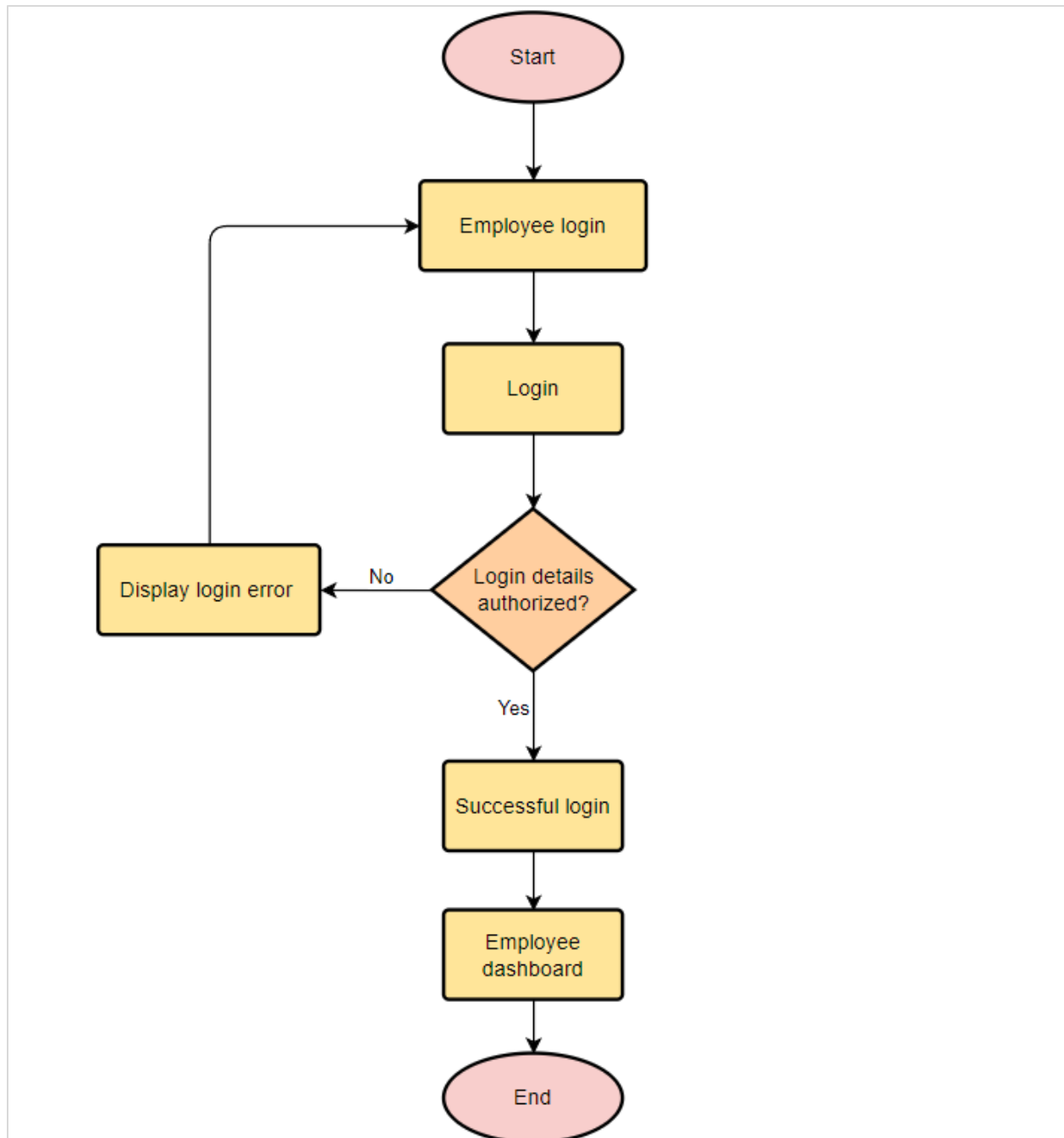


Figure 9: Employee user login - Flowchart

Advanced employee functionalities can be illustrated using an activity diagram as follows;

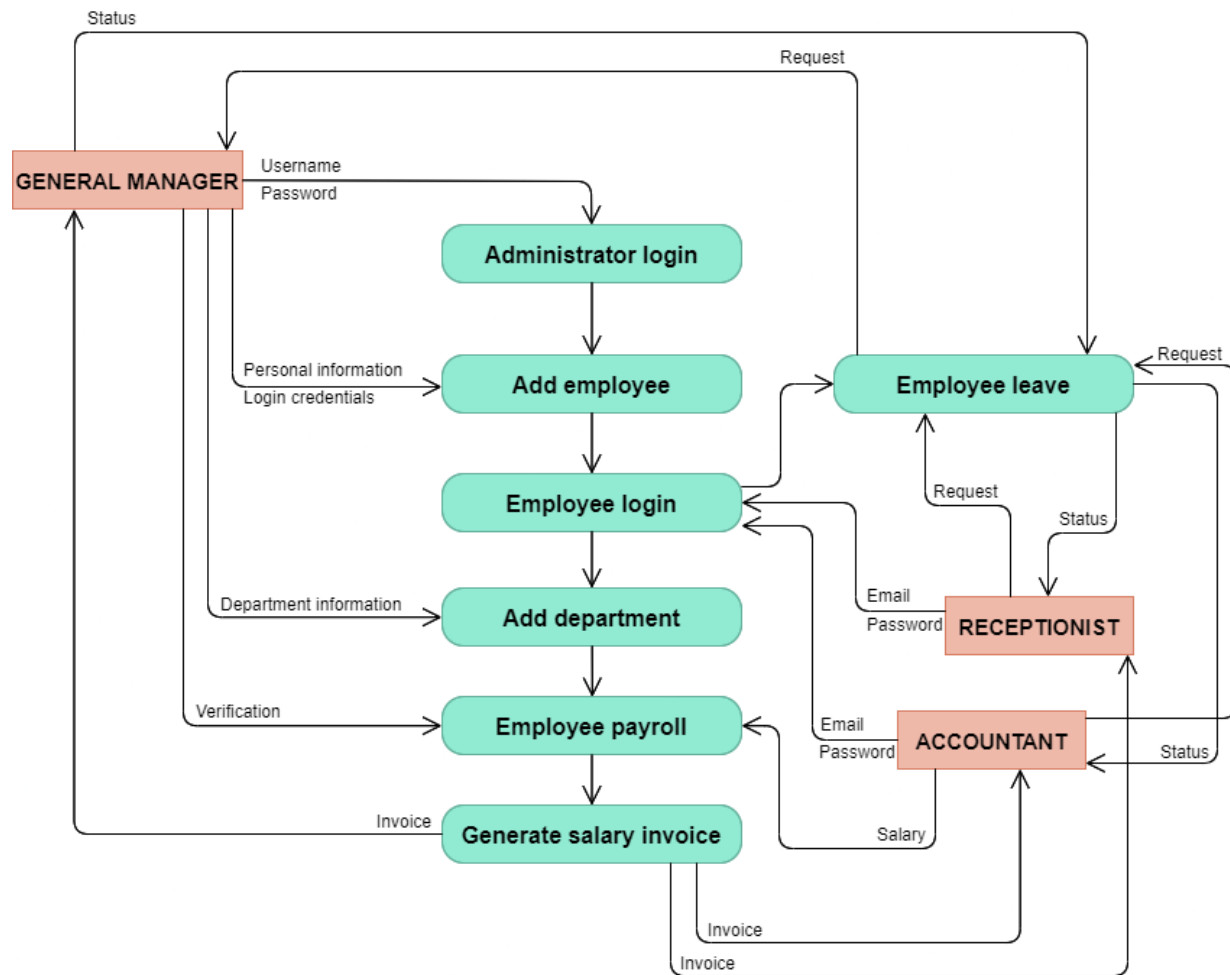


Figure 10: Employee functionalities - Activity diagram

Reserving hotel rooms through the reservation system is also an essential task of the application, which would be done by guest users. The guest can initially search for an ideal available room using the system and then reserve an ideal room after entering reservation details.

The reservation process done by the guest user can be illustrated using a flowchart as follows;

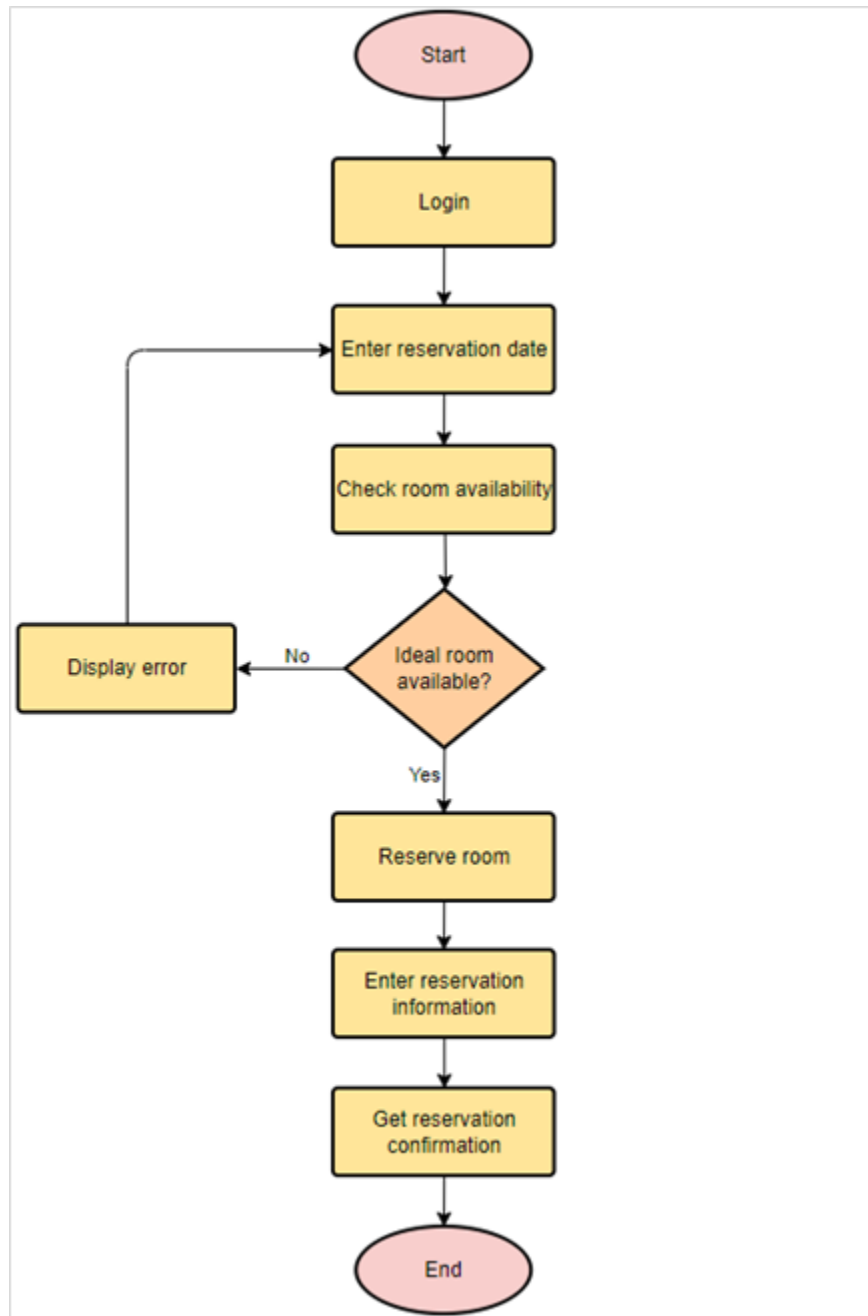


Figure 11: Room reservation - Flowchart

The reservation process can be further illustrated using a activity diagram as follows;

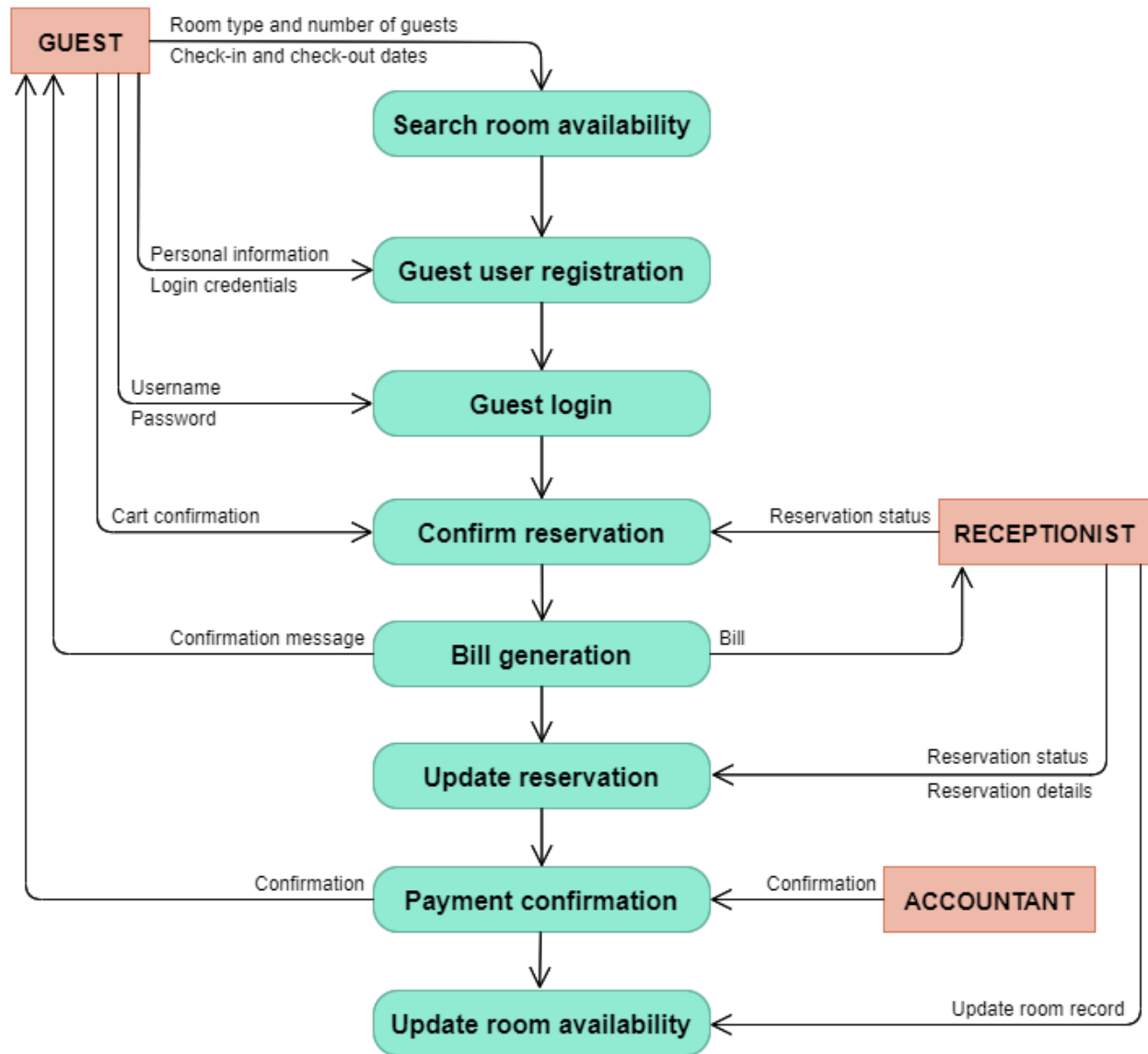


Figure 12: Room reservation - Activity diagram

3.2 System Overview

Primarily, the major aspect of any software application would be the types of users who access the system functionalities, based on their level of the permissions.

The current project's major users can be divided into four types;

01. General Manager
02. Accountant
03. Receptionist
04. Guest

Considering security, the entire reservation and centralized management system would be secured using a user login page, where all system users would be expected to have a username or email along with a unique password allocated to them as a fundamental security mechanism. Users would be unable to access any system functionality without first creating an account. This approach ensures that only those who are authorized have access to the system's contents. Moreover, the system user roles based on the functional accessed allowed for every user depends on the user type of the hotel in order to avoid risky vulnerabilities to the system security.

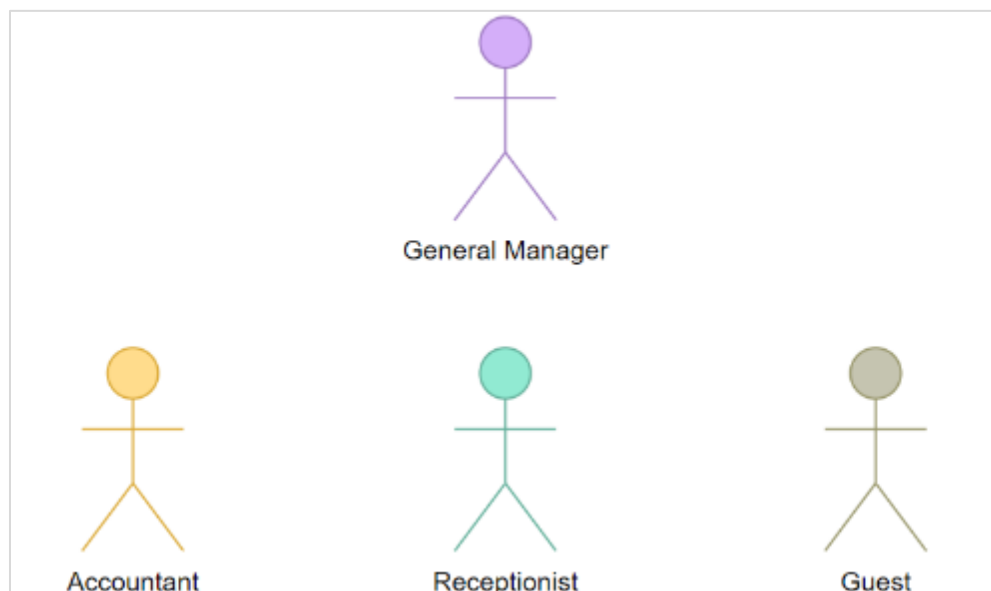


Figure 13: User Roles

Security Mechanism:

Furthermore, the web application users would be able to restrict the functions of the online reservation and centralized management system depending on the specific user role and permission. This approach has been implemented for application development in consideration of application data privacy and data security. In order to avoid risky application vulnerabilities, only the system administrators should have complete access to the entire system functionalities. The application's user role-based functionalities are listed below;

Table 8: Security mechanism - User type-based functionalities

User Type	Permitted Functionalities
General Manager	<ul style="list-style-type: none">• Add / View / Edit / Delete room type records.• Add / View / Edit / Delete room records.• Add / View / Edit / Delete reservation records.• Generate reservation income reports.• Add / View / Edit / Delete user records.• Add / View / Edit / Delete hotel department records.• Add / View / Edit / Delete employee records.• View / Edit employee leave request records.• Add / View employee payroll records.• Generate employee salary invoice.
Employee	<ul style="list-style-type: none">• Add / View / Edit / Delete room type records.• Add / View / Edit / Delete room records.• Add / View / Edit / Delete reservation records.• Generate reservation income reports.• Add / View employee leave request records.• View employee payroll records.• Generate employee salary invoice.
Guest	<ul style="list-style-type: none">• Add / View / Edit guest user account.• Add / View room reservation.• Generate room reservation receipt.

The advanced system functional procedures and user-based permissions could be illustrated using a context diagram. Context diagrams illustrate the flow of data within the system based on different users. The context diagram for the current web application project is shown below.

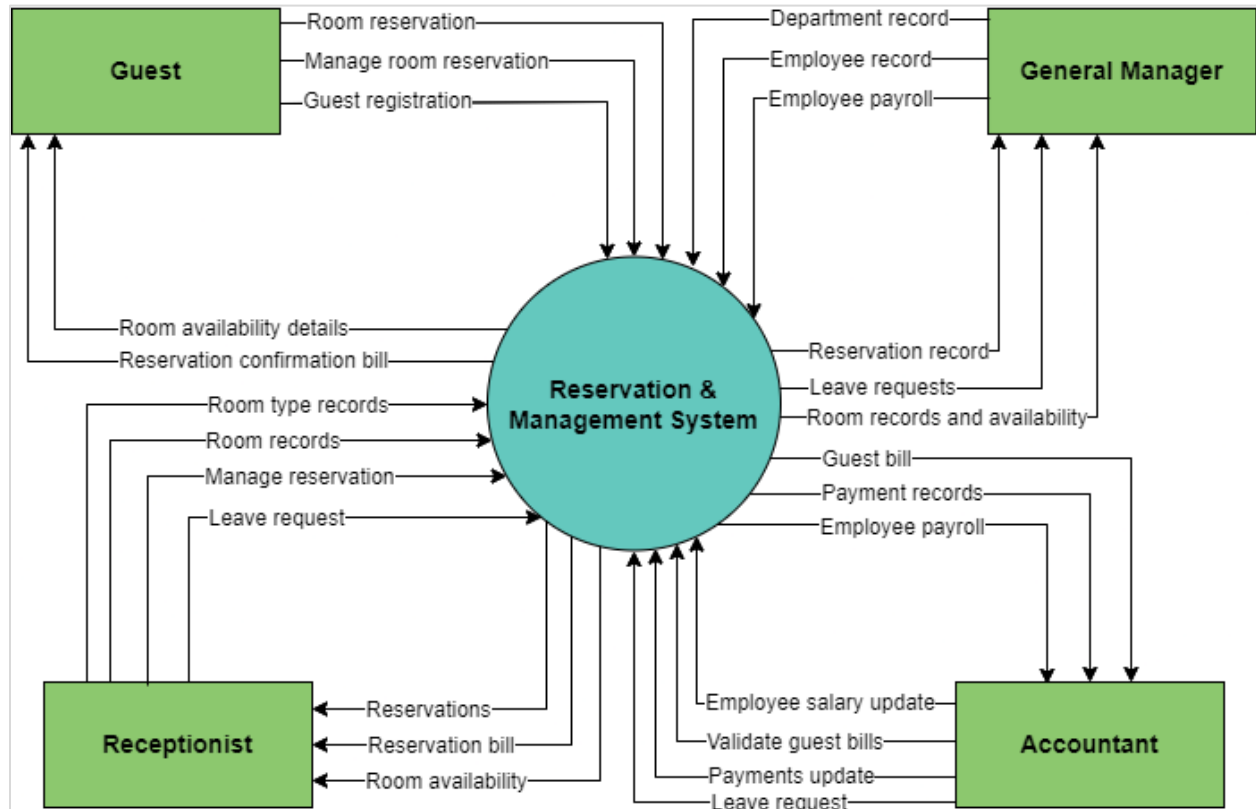


Figure 14: Context diagram

User Interface (UI) Design:

A perfect user interface is designed according to a specific step-by-step design procedure. User experiences are also integrated during these processes of implementing the design. The procedure initially starts from the step of planning, ending in final design. The process of user interface design could be illustrated using methodologies such as prototypes or wireframes.

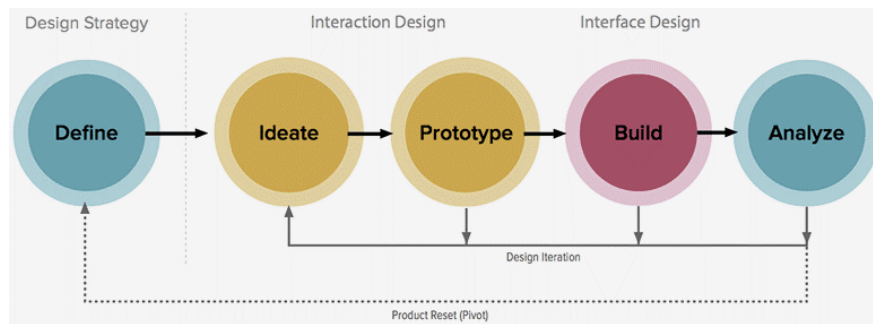


Figure 15: User interface design process

Wireframes:

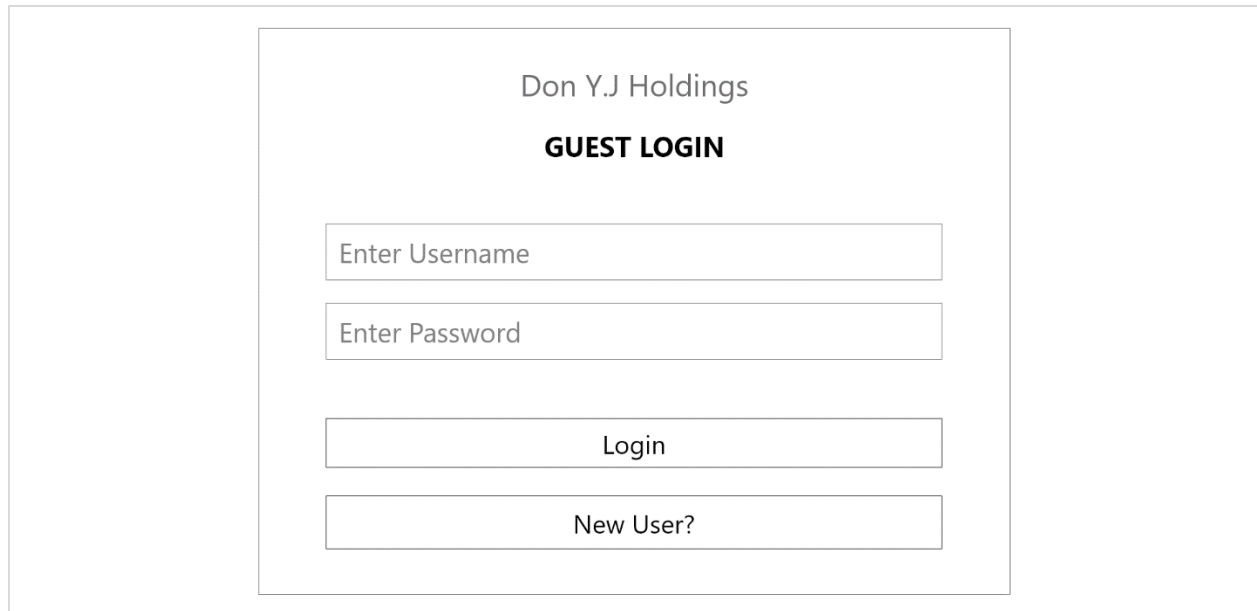
Wireframes are the consistent type of an interface sketch. Wireframes are also known as the skeleton of a user interface. It is a low-fidelity design type, which mostly includes the content, features and functions of the user interface of an application. However, visual designs and contents are not mentioned in the wireframe. The wireframe only displays the rendering of the interface, which mostly includes design architecture. (Costa, 2019)



Figure 16: Wireframes

The wireframe designs for the current application project are mentioned below.

Guest login wireframe:



The wireframe shows a central login box with a title 'Don Y.J Holdings' and a subtitle 'GUEST LOGIN'. Below the subtitle are four input fields: 'Enter Username', 'Enter Password', 'Login', and 'New User?'.

Don Y.J Holdings

GUEST LOGIN

Enter Username

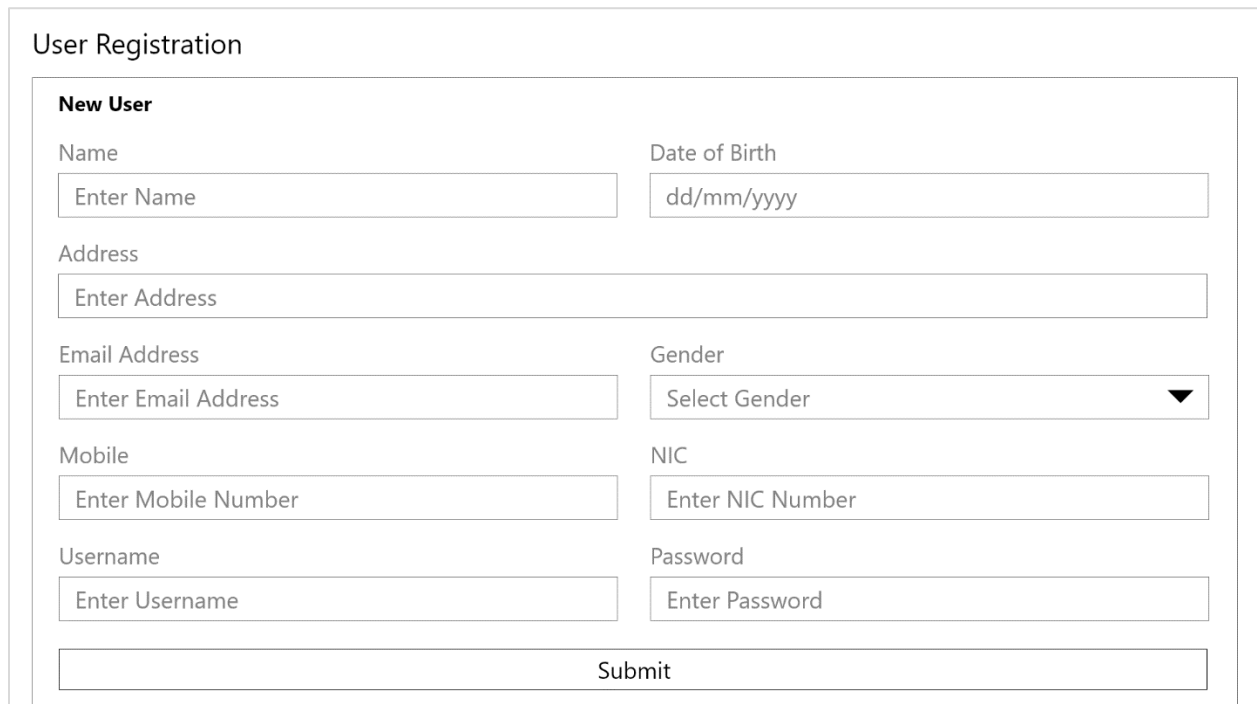
Enter Password

Login

New User?

Figure 17: Wireframe - Guest Login

Guest registration wireframe:



The wireframe shows a registration form titled 'User Registration'. It includes a 'New User' section with fields for Name, Date of Birth, Address, Email Address, Gender, Mobile, NIC, Username, and Password. A 'Submit' button is at the bottom.

User Registration

New User

Name Date of Birth

Enter Name dd/mm/yyyy

Address

Enter Address

Email Address Gender

Enter Email Address Select Gender ▼

Mobile NIC

Enter Mobile Number Enter NIC Number

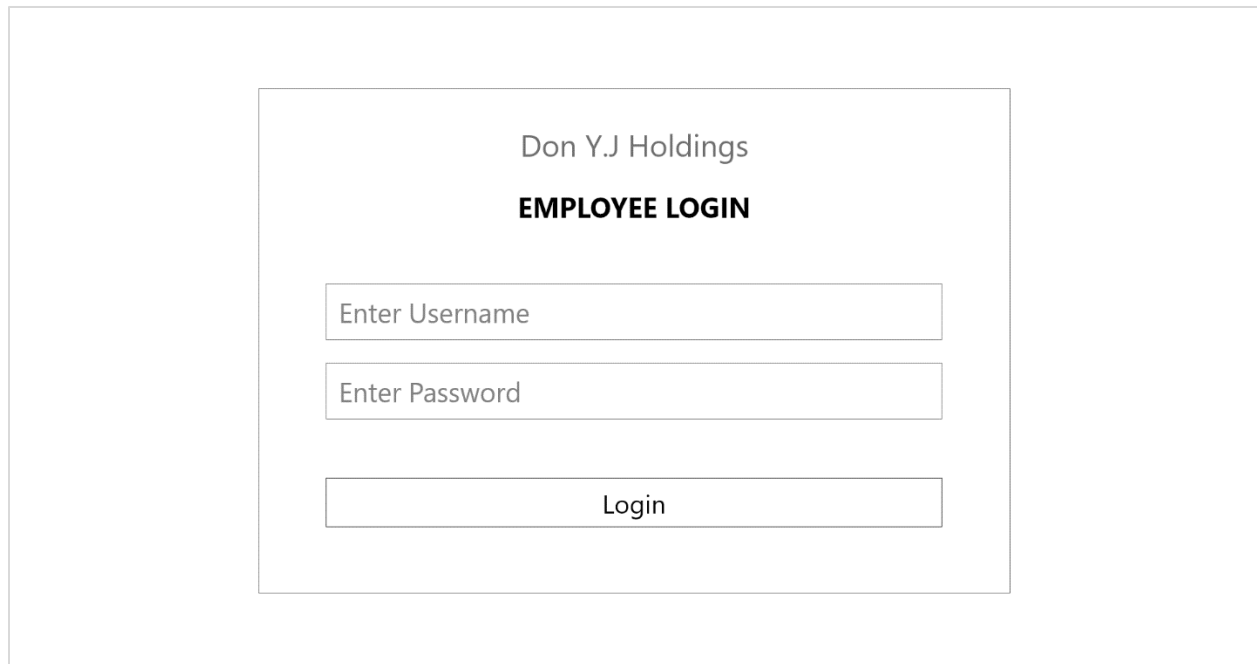
Username Password

Enter Username Enter Password

Submit

Figure 18: Wireframe - Guest Registration

Employee login wireframe:



The wireframe shows a login interface for Don Y.J Holdings. It features a central box with the company name and title, followed by input fields for username and password, and a login button.

Don Y.J Holdings

EMPLOYEE LOGIN

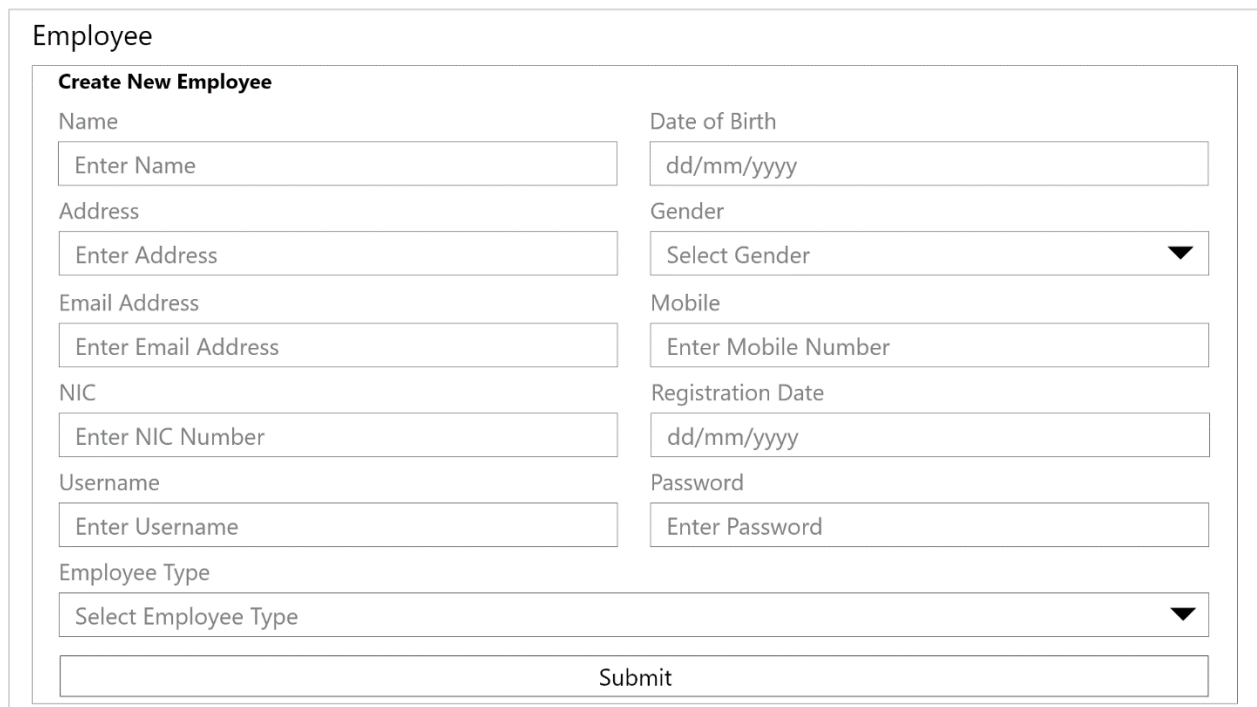
Enter Username

Enter Password

Login

Figure 19: Wireframe - Employee Login

Employee registration wireframe:



The wireframe shows an employee registration form titled 'Employee' with a sub-header 'Create New Employee'. It contains multiple input fields for personal and professional details, a dropdown for gender, and a dropdown for employee type, all leading to a submit button.

Employee

Create New Employee

Name

Enter Name

Date of Birth

dd/mm/yyyy

Address

Enter Address

Gender

Select Gender ▼

Email Address

Enter Email Address

Mobile

Enter Mobile Number

NIC

Enter NIC Number

Registration Date

dd/mm/yyyy

Username

Enter Username

Password

Enter Password

Employee Type

Select Employee Type ▼

Submit

Figure 20: Wireframe - Employee Registration

Hotel reservation system home page wireframe:

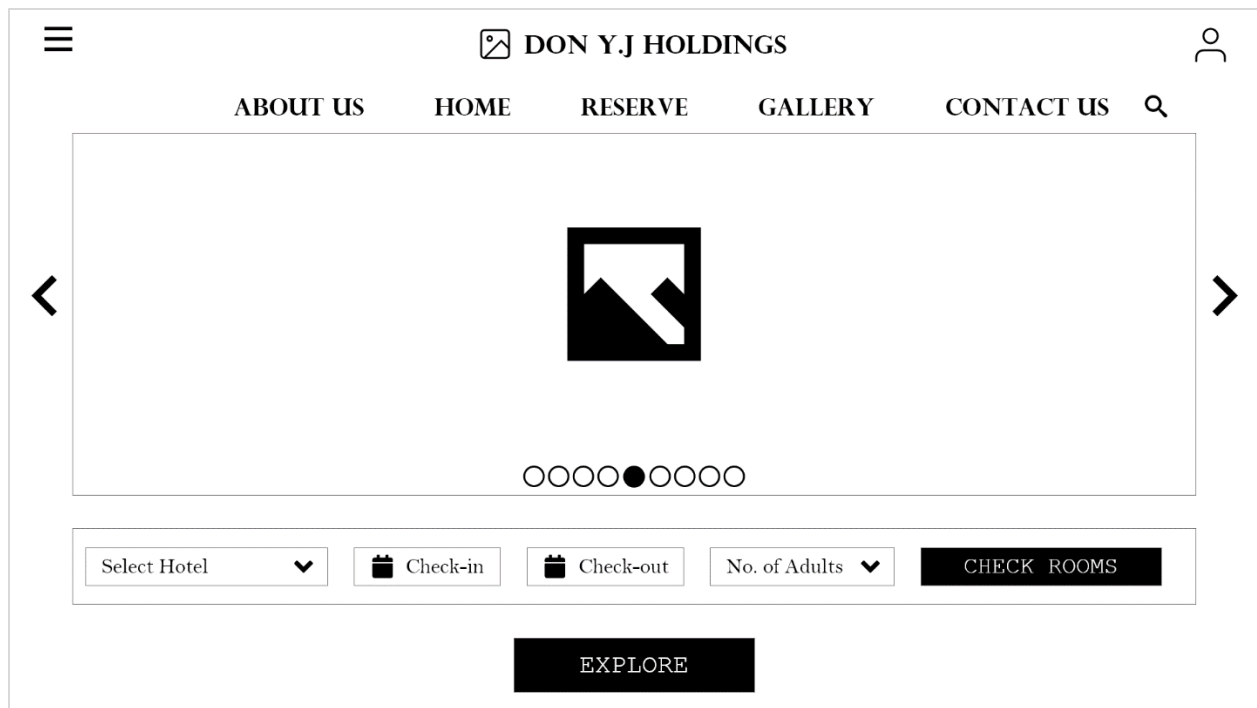


Figure 21: Wireframe - Hotel reservation system home page

Hotel reservation system gallery wireframe:

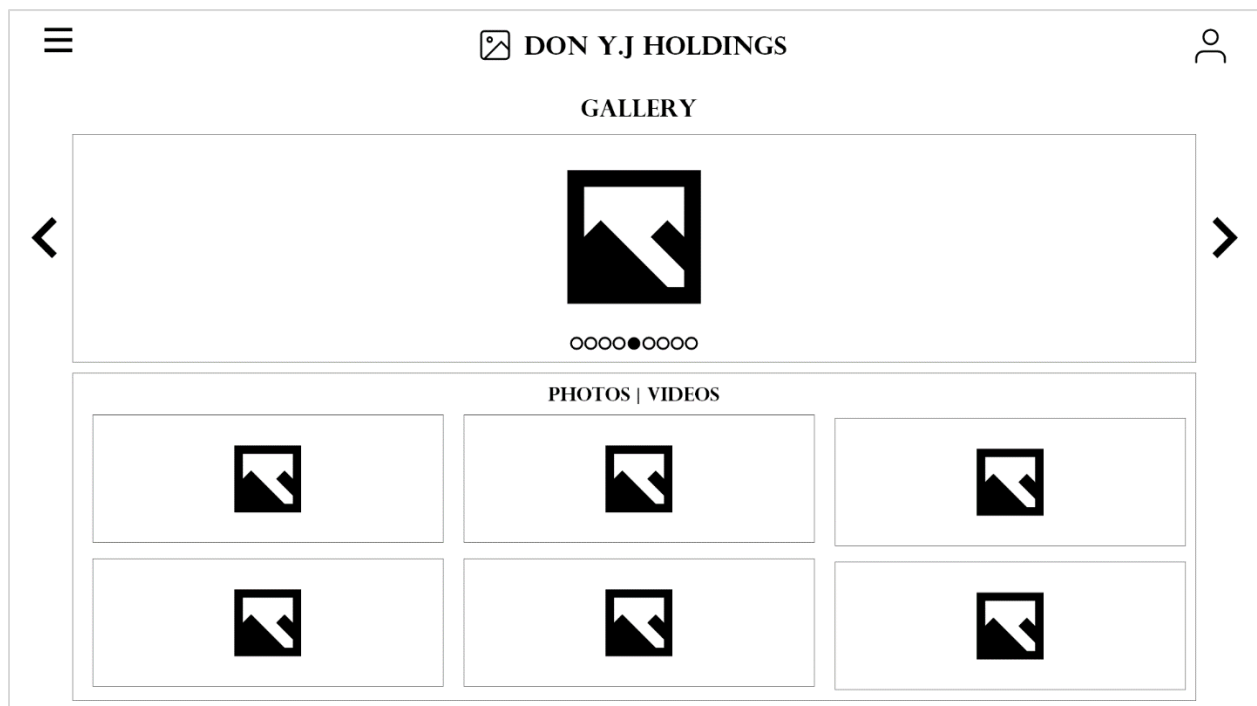


Figure 22: Wireframe - Hotel reservation system gallery

Hotel reservation search page wireframe:

Header: DON Y.J HOLDINGS

Navigation: ABOUT US, HOME, RESERVE, GALLERY, CONTACT US

Form Fields:

- Select Hotel
- Check-in
- Check-out
- No. of Adults
- CHECK ROOMS
- Select Reservation Type
- Reservation Type
- Reservation type description
- CONTINUE

Figure 23: Wireframe - Hotel reservation search page

Hotel reservation details confirmation page wireframe:

Header: DON Y.J HOLDINGS

Navigation: ABOUT US, HOME, RESERVE, GALLERY, CONTACT US

Form Fields:

- First Name
- Middle Name
- Last Name
- Address
- Country
- State
- City
- Postal Code
- Email
- NIC
- Mobile
- Alternate Mobile
- CONTINUE

Figure 24: Wireframe - Hotel reservation search page

4. Product Implementation

The proposed web application system will be divided in two sections during development, as reservation system and a centralized management system, which will be eventually linked with appropriate user permissions to access functions. The reservation system would be a web application system where hotel customers can be updated with company information and also book hotel reservations. The management system will be a web application which can only be accessed by the company staff to store and retrieve data regarding the company and its users.

The picked development languages and technical resources are listed below.

Table 9: Development technologies

Front-End Development	HTML, CSS, JavaScript and Bootstrap Framework.
Back-End Development	PHP, JavaScript and CodeIgniter Framework.
Database	MySQL.
Local Web Server	XAMPP
Source Code Editor	Visual Studio Code / Sublime Text

Technical resources required to configure the application are;

Table 10: Required technical resources

Hardware requirements	<ul style="list-style-type: none">• Minimum Pentium 4 or newer processor.• Minimum 4GB of RAM.• Minimum 500GB of HDD storage.• Stable internet router connection.
Software requirements	<ul style="list-style-type: none">• Running Windows 8/macOS Sierra 10.12 or newer system operating system.• Internet browser, Google Chrome recommended.
Application resources	<ul style="list-style-type: none">• Web application domain name.• Web server with database system for application hosting.

The descriptions of the different selected tools and technologies selected to develop the web application are mentioned below.

Web Application:

A Web Application is an application program, where the program is hosted directly to the internet using a server, where all the application files will be stored. A web application can be accessed using a web browser. Web applications are popular for their easily accessibility by anyone, using any device. When a user visits a web application using a browser, the program files are viewed directly from the stored servers. (Rouse, 2019)

HTML:

Hyper-Text Markup Language (HTML) a language used to design and structure the interfaces of a website. The language instructs the web browser on how the website content must be displayed. HTML mostly structures text and multimedia components on the website pages. HTML uses tags for the inclusion on content on a site, and filed with ‘.html’ extension. HTML can be created and edited using basic text editors. (Rouse, 2020)

CSS:

Cascading Style Sheets (CSS) is an example for style sheets, which is usually integrated with HTML language, focusing on the website design. Simply, CSS helps in formatting HTML pages of the website. CSS can be linked to HTML by declaring its file location or called using a ‘Style’ tag. CSS formatting can be used to change text colour, background colour or content alignments on the page, whereas the file extension is ‘.css’. (Rouse, 2018)

Bootstrap:

Bootstrap is an open-source CSS framework, implemented for website front-end interface developments. The framework includes various features to enable responsive and mobile-site websites. Responsive sites mean that the website orientation wouldn’t format on any device dimensions. Bootstrap also includes usual CSS style designs. (Rouse, 2017)

JavaScript (JS):

JavaScript (JS) is a programming language integrated to implement logics to websites and its components. JS is a popular browser-based language, but also works well with server programming. HTML is basically a static language, which doesn’t allow logics. So, JavaScript is added to work out the website logics. Simply, any changed during the runtime of a website can be set using JavaScript, such as progress bars or animations. (Rouse, 2018)

PHP:

PHP, also known as Hypertext Pre-Processor, is a script language, mostly used on server-side programming. PHP works with installed servers, including web servers mainly. PHP is added along with HTML language, but is executed on the server, mainly for database functions. When a user interacts with a PHP function, the web server calls the PHP script for the specific function to be executed. A PHP script file is provided with '.php' extension. (Rouse, 2006)

Laravel:

Laravel is an open-source PHP framework, to implement server functions on the website back-end development. Laravel simplifies PHP functions and is easy to learn, which saves a great amount of development time. Laravel also contributes in organizing of website resources using interfaces and namespaces. It also includes features such as testability functions, database schema builder and template engines with predefined blocks. (TutorialsPoint, 2018)

MySQL:

MySQL is a relational database system, developed by Oracle. This is the most popular database system, with SQL functions. MySQL is an open-source database. (TutorialsPoint, 2018)

Visual Studio Code:

Visual Studio Code is an open-source and free software source-code editor developed by Microsoft. The editor supports various development and testing features along with a user-friendly interface environment, ideal for desktop and web implementations.

Sublime Text:

Sublime Text is an advanced text editor tool. The tool has a simple interface, including advanced features such as file type recognition, simultaneous edits, code folding, syntax highlight, auto indentations, code mini-map and plugin package support. Sublime text is a premium paid tool, but could be used with an unlimited evaluation period for free. (Schäferhoff, 2017)

XAMPP:

XAMPP is an open-source and free local web server solution software. The server supports development with Apache HTTP server, PHP, Perl, MySQL and MariaDB. XAMPP stands for (Cross-Platform(X) Apache MySQL PHP Perl). The server lets web developers to even test advanced web applications locally without the use of any online tool.

Development Methodologies:

Considering Don Y.J company's requirements, "Agile" system development methodology would be an ideal choice. The agile methodology refers a practice that the desired application should go through continuous revisions within its development and testing stages. This promotes a situation where the system is constantly checked for errors and inconsistencies over every major phase in the development stage. Agile provides reliable assistance among developers and clients to develop and deliver systems in a proper timeline, while having efficient responses for any functional changes or requests. (Hamilton, 2021).

"Scrum" model, which promotes an organized development procedure will be the current project's development model. Scrum promotes to divide the project into multiple sections called sprints, and then upgrade these sprint outcomes until the requirements are met (Lutkevich, 2021). Thus, the application can be regularly tested and reworked for better effectiveness as it would be possible to constantly revise the application. This also ensures that the application would include proper user experience and proper user functionalities for efficient usage.

The entire project is divided into seven major phases, which are listed below.

01. Research.
02. Data collection.
03. Architecture Design.
04. Interface Design.
05. Development.
06. Quality Assurance.
07. Documentation.

Project Milestones:

01. Comprehensive project research.

- Focusing on the scenario and required technological aspects.

02. Gathering system requirements.

- Focusing on how the application should interact varyingly among different user types such as customers, employees and higher authorities.

03. System architectural design.

- Designing system substructure based on requirements, assisted with software diagrams.

04. System elements design.

- Designing the visual elemental contents including application interfaces.

05. Development.

- The application will be divided into sub-sections, which will be developed in an efficient order, which ensures that the development process will go through a smooth flow.

06. Quality assurance.

- Any divided development sections will be linked together with the main application. If any errors are identified, it will be moved back to the development stage for revision.

07. Execution and evaluation.

- The application will undergo functional evaluation to study and report how multiple system architectures work together after deployment in the environment.

08. Final documentation.

- To be designed including the evaluation reports of project approaches taken, quality assurance procedures and the system user manuals.

Project Deliverables:

01. Online hotel reservation and centralized management web application system.

02. Final documentation.

03. System user guide.

Project Workplan:

ID	Task Name	Start	Finish	Duration	1/11/2021	1/1/2022				1/4/2022
					1/11/2021	1/12/2021	1/1/2022	1/2/2022	1/3/2022	1/4/2022
1	☐ SPRINT 01	1/11/2021	29/11/2021	29.0 d.						
2	Sprint Planning	1/11/2021	2/11/2021	2.0 d.						
3	Research	2/11/2021	15/11/2021	14.0 d.						
4	Data Collection	16/11/2021	29/11/2021	14.0 d.						
5	☐ SPRINT 02	30/11/2021	3/1/2022	35.0 d.						
6	Sprint Planning	30/11/2021	1/12/2021	2.0 d.						
7	Architecture Design	1/12/2021	14/12/2021	14.0 d.						
8	Interface Design	15/12/2021	3/1/2022	20.0 d.						
9	☐ SPRINT 03	5/1/2022	5/3/2022	60.0 d.						
10	Sprint Planning	5/1/2022	6/1/2022	2.0 d.						
11	Development	7/1/2022	5/3/2022	58.0 d.						
12	☐ SPRINT 04	6/3/2022	31/3/2022	26.0 d.						
13	Sprint Planning	6/3/2022	7/3/2022	2.0 d.						
14	Quality Assurance	8/3/2022	31/3/2022	24.0 d.						
15	☐ SPRINT 05	1/4/2022	30/4/2022	30.0 d.						
18	Sprint Planning	1/4/2022	2/4/2022	2.0 d.						
17	Documentation	3/4/2022	30/4/2022	28.0 d.						

Figure 25: Project workplan

Project Delivery Structure:

Considering software projects targeted towards clients generally, it is a professional practice that the entire project deliverables should be arranged and categorized in a proper manner, so that the new users could still learn the software functionalities. This project even includes a user guide as a final deliverable, so that any new user in the client company would be able to understand the functionalities of the system and how they are operated.

Coding Principles:

Coding principles are a pretty important concept as programmers, because they determine the final execution of any software system. Developers use certain principles and standards while software development in order to maintain a specific standard in the software codes. Especially in projects where multiple developers work together, coding principles would be an essential concept, as they assist developers understand the programming of the other members of the team.

The coding principles considered to be implemented in the current project are described below;

01. Clean Code

Clean code principle is a pretty straight forward concept, focusing on maintaining a clean and aligned code. Clean code is a professional manner of maintaining an organized coding structure, so that the contents of the codes would be easily understandable any time. For example, the snippet shown below illustrates how cleaner code is easier to understand for developers;

02. Code Commenting

Code commenting refers to a practice of describing certain functionalities in the application on what they are programmed towards. Eventually, whenever the programmer has to revise or evaluate the code, it would be easier to understand the purpose of the code components.

03. Code Indentation

Code indentation is a simple concept to maintain even amounts of space within code rows and columns, so that the entire program looks cleaner. This allows the programmer to separate different coding components separately, so they are more understandable later.

04. Structured Source Code

A software with structured source code means that the software files would be categorized efficiently in a proper order. This ensures that every component of the software is properly separated and listed effectively to understand the code structure in a better manner.

05. Naming Convention

While naming software components, it is essential that the programmer would be able to remember or even at times guess the names. To achieve this, developers use universal naming conventions with similar alphabetical contents to name components and identify them easier.

Software Architectural Design Pattern:

Software architectural design patterns are considered as certain stated standards or principles implemented by developer to ensure a straightforward and efficient application development procedure. Architectural design patterns are typically a widespread notion within object-oriented based software programmers. Design patterns are mostly implemented to avoid common and recurring issues and bugs during development. A typical design pattern would seem like an unfinished code, but could be used as a universal template to avoid development issues throughout the entire software application project process.

In consideration of the current project requirements, the MVC (Model, View, Controller) architectural pattern would be the most suitable architectural design pattern selection, suitable for Don Y.J Holding's online reservation system and a centralized management system. The MVC design pattern would promote code modularity and maintainability for the development procedure. MVC is considered as a popular and widespread architectural design pattern that could be implemented to a wide range of programming languages as well as related software frameworks. The intended web application would consist multiple types of software components that would be required to connect and execute together. These components would be required to execute a wide range of varying business logics and database connection events.

The front-end based visual interface screens for the functionalities of the web application that the end user is displayed is required to depend on the user permissions and roles. Considering essential business logics and standards the system database functions, application screens and the specific software architecture rationally separates the system source code and functionalities. Programmers would be able to accomplish the final application codes with less difficulties as there are an efficient modularity. Moreover, even multiple developers would have the ability to develop and update components of the system source code simultaneously, which decreases the development period while having an efficient team productivity.

Development Folder Structure:

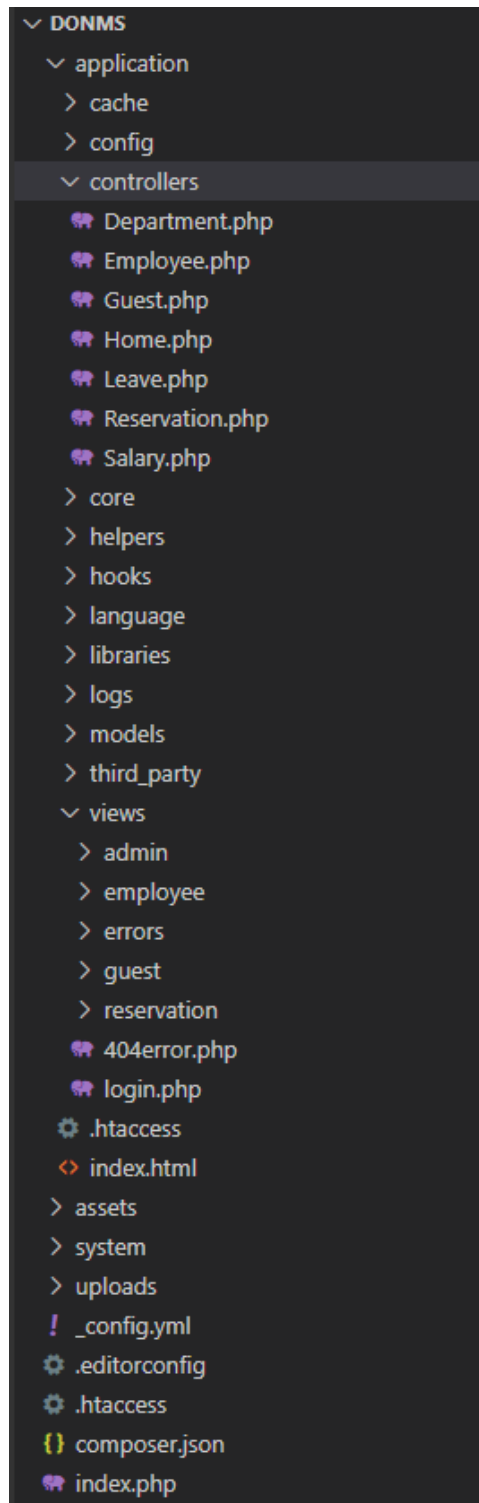


Figure 26: Development folder structure

Interfaces:

Don Y.J Holdings application Home Page:

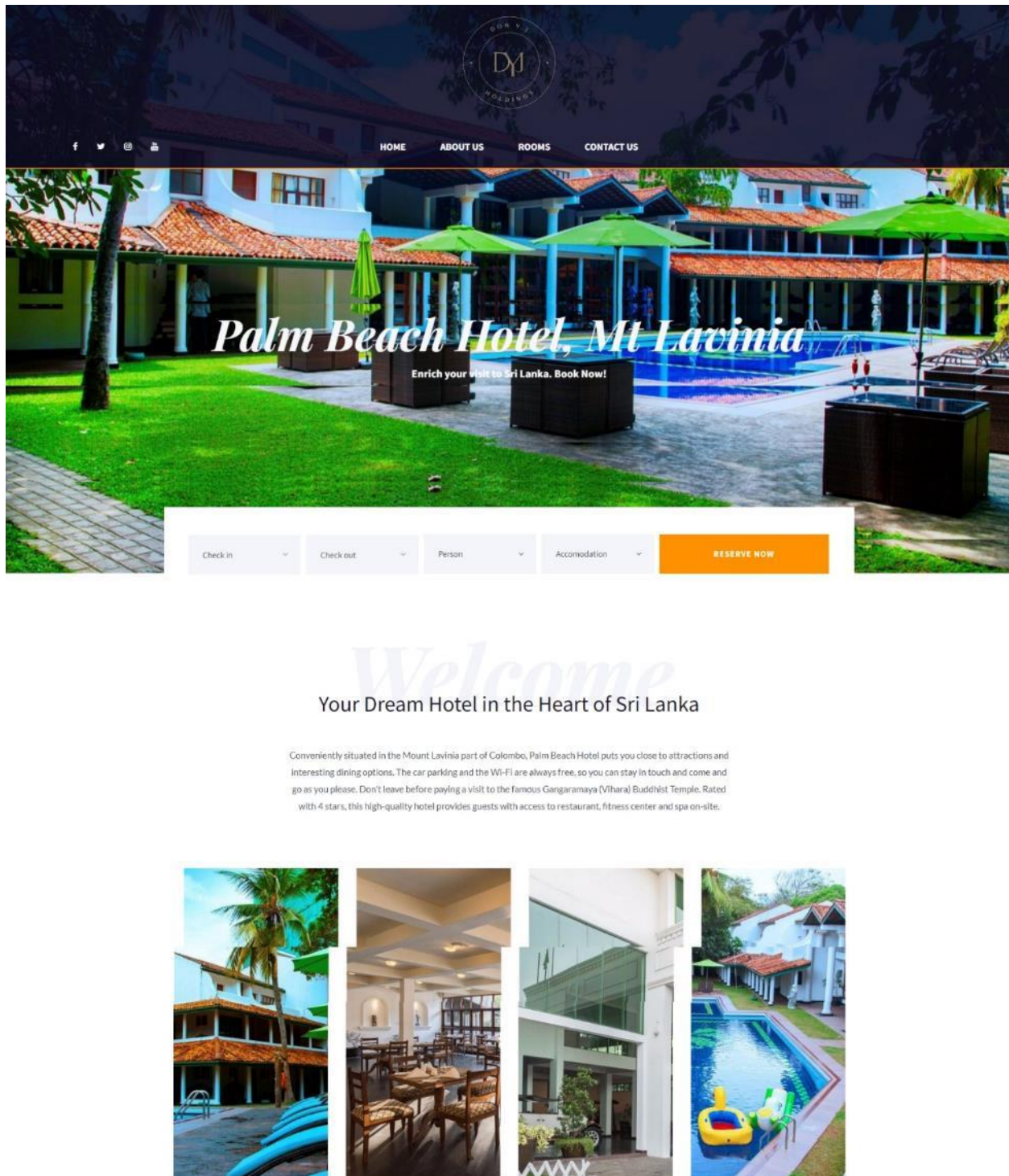


Figure 27: Interfaces - Homepage

Room Rates:

The screenshot displays the website for Palm Beach Hotel, Mt Lavinia. The header features a dark blue background with a circular logo containing the letters 'DM' and the word 'HOTELS'. Below the logo are social media icons for Facebook, Twitter, Instagram, and YouTube. A navigation menu includes links for HOME, ABOUT US, ROOMS, and CONTACT US. The main banner shows a vibrant image of the hotel's outdoor pool area with green umbrellas and lounge furniture. The text 'Palm Beach Hotel, Mt Lavinia' is overlaid in a large, elegant font, with a smaller tagline 'Enrich your visit to Sri Lanka. Book Now!' below it. A reservation form is positioned below the banner, featuring dropdown menus for 'Check in', 'Check out', 'Person', and 'Accommodation', followed by an orange 'RESERVE NOW' button. Below the form, three room options are presented in white cards with orange accents. Each card includes a room image, the room name, guest and room availability information, the nightly rate, and a 'RESERVE' button.

Room Type	Number of Guests	Rooms Left	Rate (USD / Night)
ROOM ONLY SINGLE ROOM	1	1	70
ROOM ONLY DOUBLE ROOM	2	1	80
ROOM ONLY TRIPLE ROOM	3	1	90

Figure 28: Interfaces - Room rates

Contact Us:

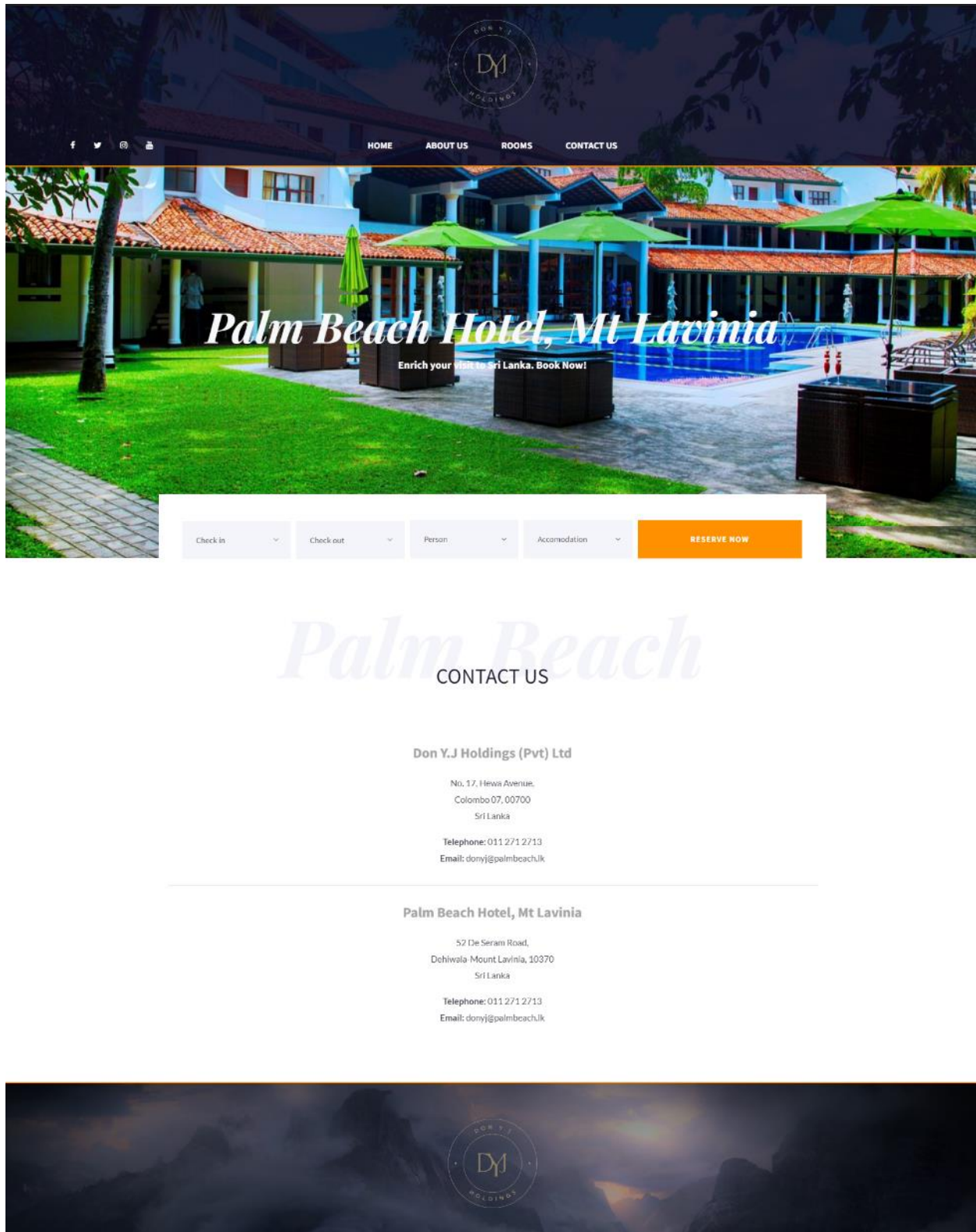
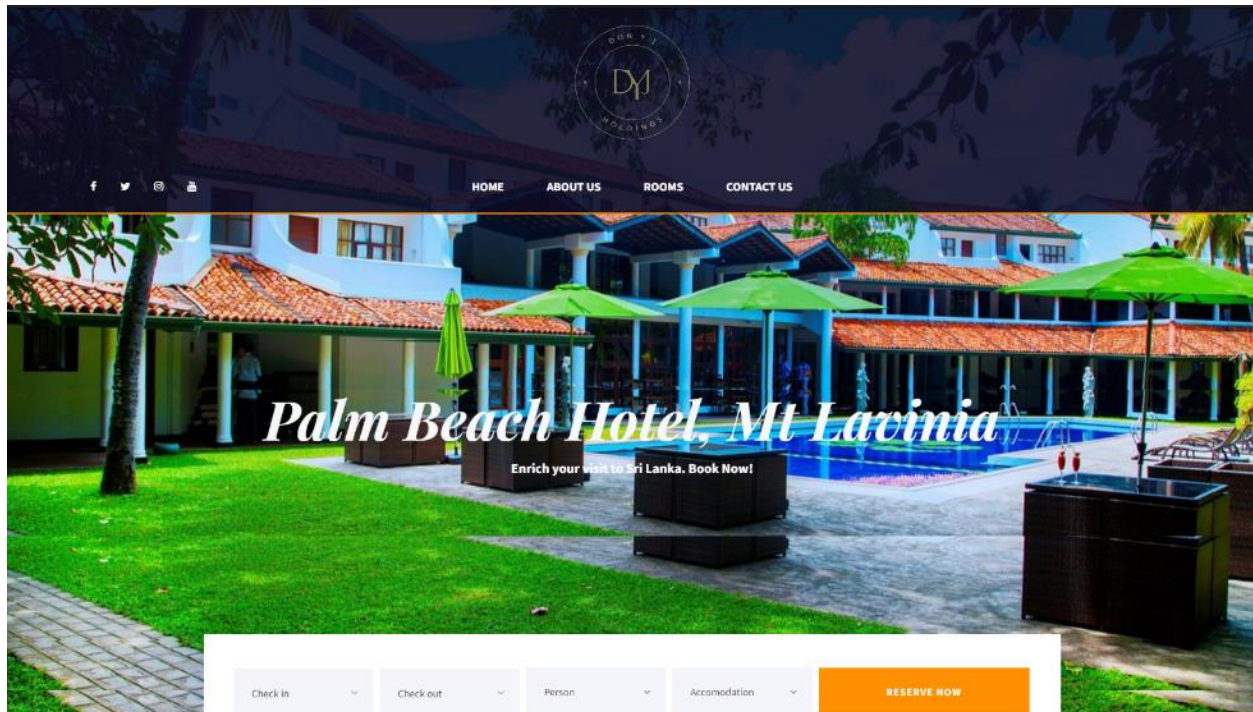


Figure 29: Interfaces - Contact Us

Reservation Cart:



Your Reservation Cart

Room	Check In	Check Out	Price	Nights	Amount	Action
ROOM ONLY	04/25/2022	04/26/2022	€70	1	\$70	Remove
BFD & BREAKFAST	04/25/2022	04/26/2022	€120	1	\$120	Remove
FULL BOARD	04/25/2022	04/26/2022	€150	1	\$150	Remove
					Total:	340 USD

[Clear Cart](#) [CONTINUE RESERVATION](#)

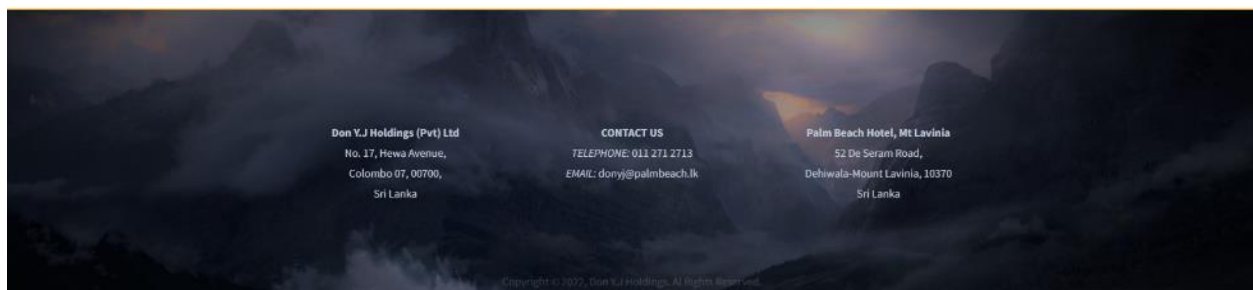
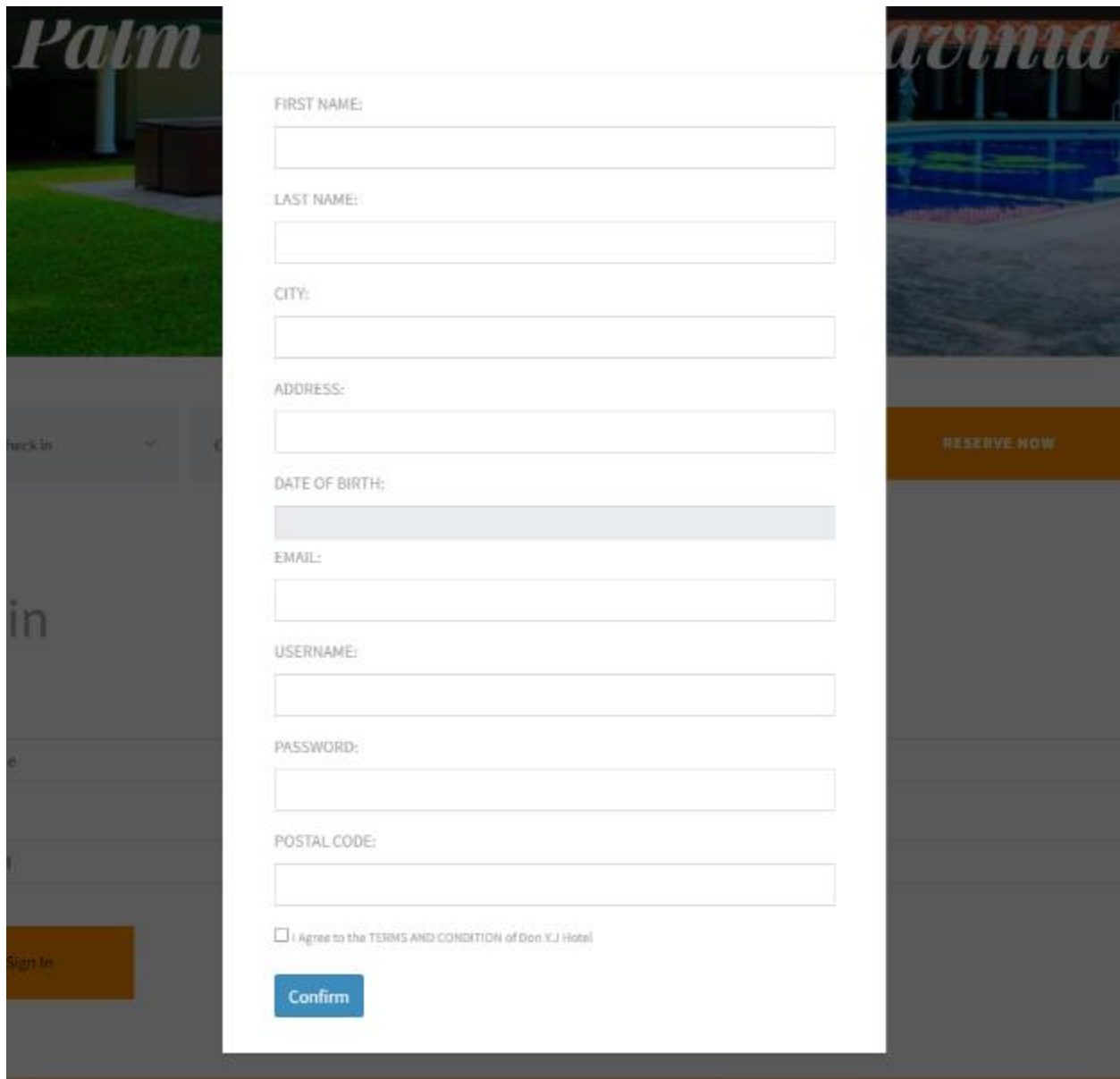


Figure 30: Interfaces - Reservation Cart

Guest Registration:



The image displays a guest registration form overlaid on a website background. The background features a large image of a swimming pool with the word "Palm" on the left and "avina" on the right. A "RESERVE NOW" button is visible on the right side of the background. The registration form is a white rectangular box with the following fields and elements:

- FIRST NAME:** A text input field.
- LAST NAME:** A text input field.
- CITY:** A text input field.
- ADDRESS:** A text input field.
- DATE OF BIRTH:** A date selection field.
- EMAIL:** A text input field.
- USERNAME:** A text input field.
- PASSWORD:** A text input field.
- POSTAL CODE:** A text input field.
- ☐ I Agree to the **TERMS AND CONDITION** of Don Y.J Hotel
- Confirm** button

Figure 31: Interfaces - Guest Registration

Guest Login:

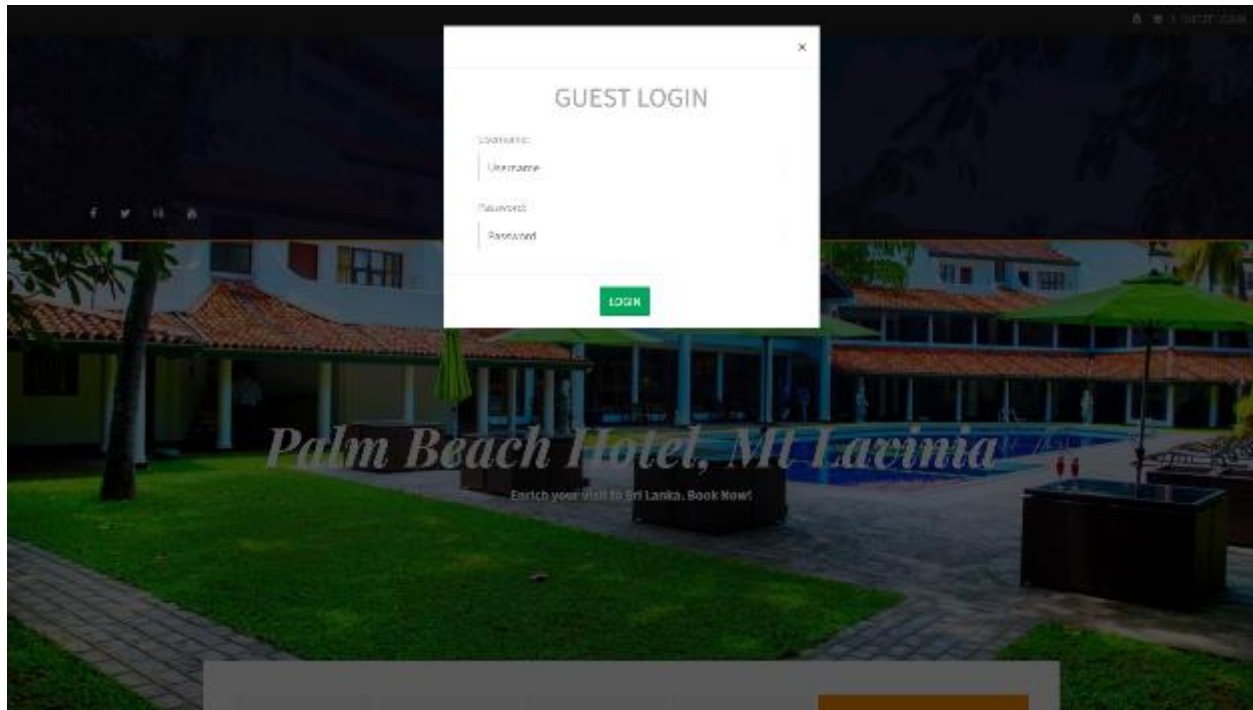


Figure 32: Interfaces - Guest Login

Alternative Guest Login:

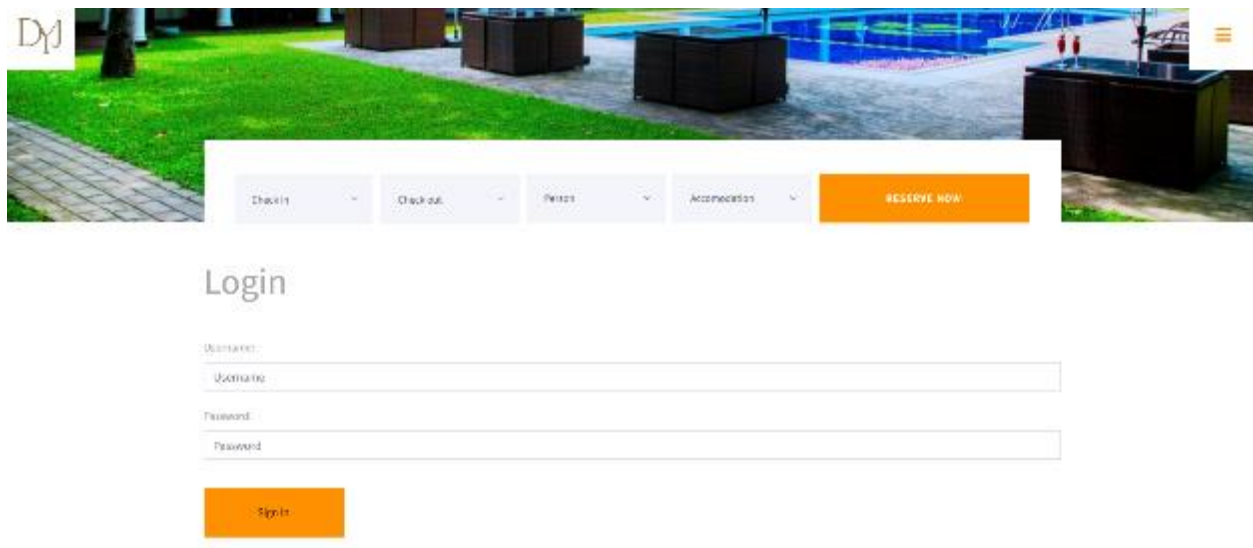


Figure 33: Interfaces - Alternative Guest Login

Confirm Reservation:



Billing Information

Booking Details					
Name: Robert Williams Address: 45, Main Street Mobile: 0714563248			Reservation Date: 04/24/2022 Reservation ID: 5467456541		
Room	Arrival	Departure	Price	Night(s)	Subtotal
HALF BOARD	04/24/2022	04/24/2022	\$168	1	\$168
FULL BOARD	04/24/2022	04/24/2022	\$150	1	\$150
FULL BOARD	04/24/2022	04/24/2022	\$210	1	\$210
					Total: 528 USD
CONFIRM RESERVATION					

Figure 34: Interfaces - Confirm Reservation

Reservation System Admin Login:

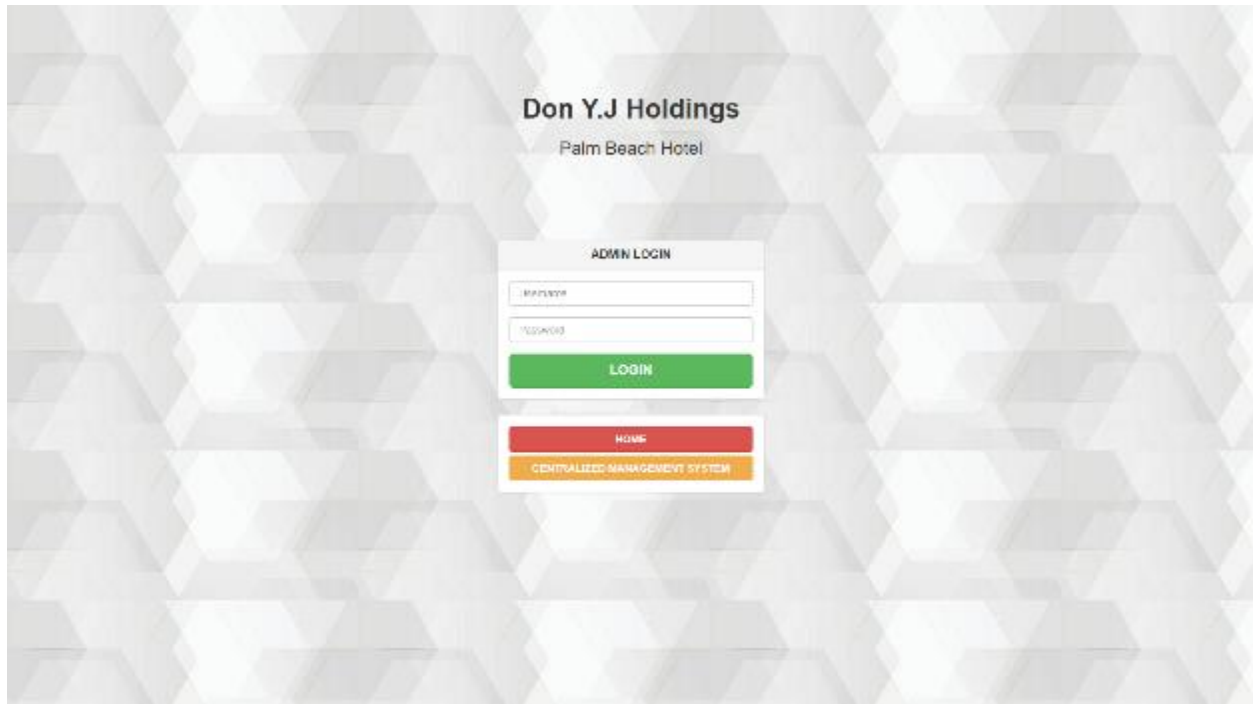


Figure 35: Interfaces - Reservation System Admin Login

Create New Room Type:

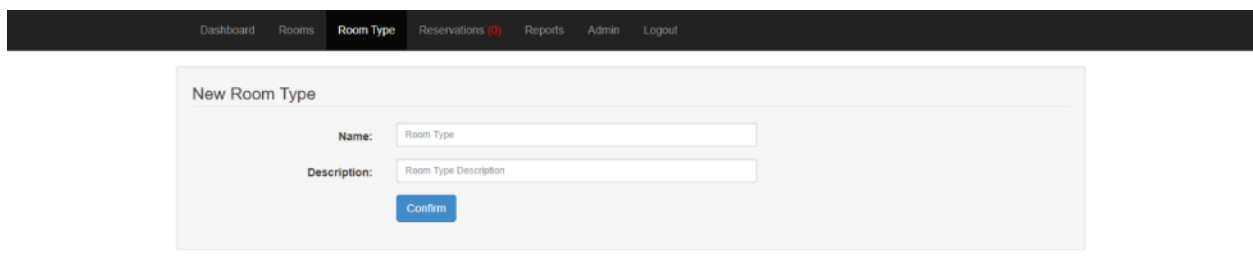


Figure 36: Interfaces - New Room Type

View Available Room Types:

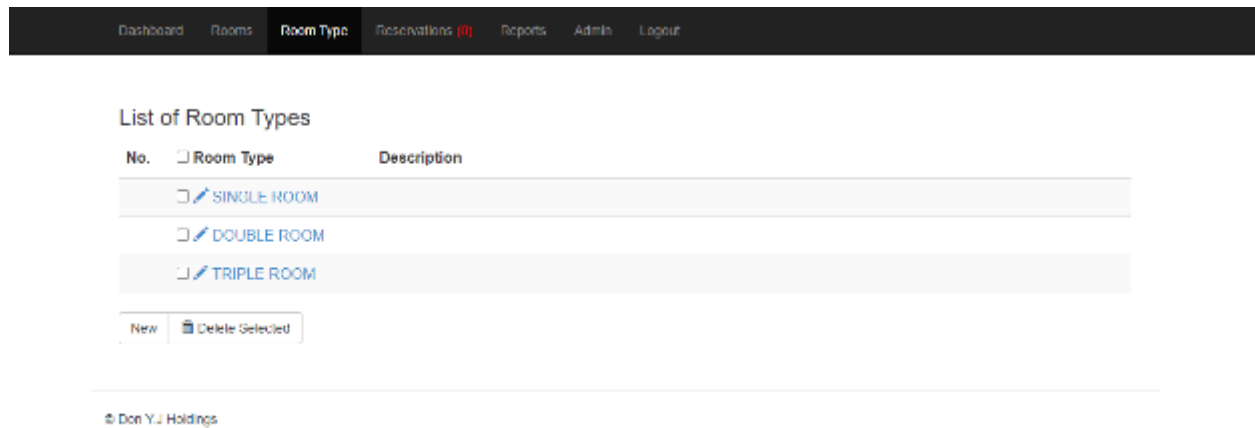


Figure 37: Interfaces - View Available Room Types

Edit Available Room Type:

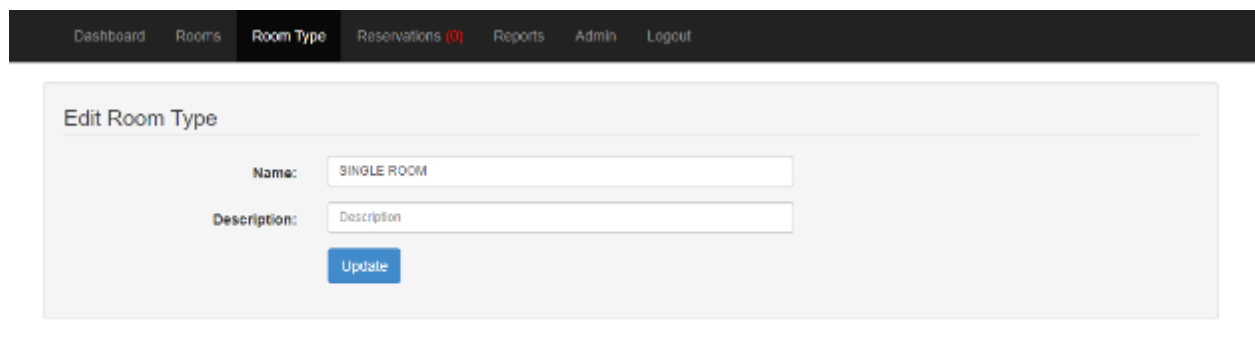


Figure 38: Interfaces - Edit Available Room Type

Create New Room:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(0\)](#) [Reports](#) [Admin](#) [Logout](#)

New Room

Name:

Room Type:

Description:

Number of Person:

Price (\$):

Upload Image: No file chosen

Figure 39: Interfaces - Create New Room

View Available Rooms:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(0\)](#) [Reports](#) [Admin](#) [Logout](#)

List of Rooms

No.	<input type="checkbox"/> Image	Room	Room Type	Person	Price
<input type="checkbox"/>		ROOM ONLY ()	SINGLE ROOM	1	\$70
<input type="checkbox"/>		BED & BREAKFAST ()	SINGLE ROOM	1	\$80
<input type="checkbox"/>		HALF BOARD ()	SINGLE ROOM	1	\$96
<input type="checkbox"/>		FULL BOARD ()	SINGLE ROOM	1	\$110
<input type="checkbox"/>		ROOM ONLY ()	DOUBLE ROOM	2	\$80
<input type="checkbox"/>		BED & BREAKFAST ()	DOUBLE ROOM	2	\$90
<input type="checkbox"/>		HALF BOARD ()	DOUBLE ROOM	2	\$122
<input type="checkbox"/>		FULL BOARD ()	DOUBLE ROOM	2	\$150
<input type="checkbox"/>		ROOM ONLY ()	TRIPLE ROOM	3	\$90
<input type="checkbox"/>		BED & BREAKFAST ()	TRIPLE ROOM	3	\$120
<input type="checkbox"/>		HALF BOARD ()	TRIPLE ROOM	3	\$168
<input type="checkbox"/>		FULL BOARD ()	TRIPLE ROOM	3	\$210

© Don Y.J Holdings

Figure 40: Interfaces - View Available Rooms

Edit Available Rooms:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(0\)](#) [Reports](#) [Admin](#) [Logout](#)

Edit Room

Name:

Room Type:

Description:

Number of Person:

Price (\$):

Upload Image: No file chosen




Figure 41: Interfaces - Edit Available Rooms

View and Manage Reservations:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(1\)](#) [Reports](#) [Admin](#) [Logout](#)

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Anna Hethers	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	<input type="button" value="View"/> <input type="button" value="Cancel"/> <input type="button" value="Confirm"/> <input type="button" value="Delete"/> <input type="button" value="Edit"/>
	Robert Williams	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	<input type="button" value="View"/> <input type="button" value="Delete"/> <input type="button" value="Edit"/>

© Don Y.J Holdings

Figure 42: Interfaces - View and Manage Reservations

View Reservation Record:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(1\)](#) [Reports](#) [Admin](#) [Logout](#)

Guest Information


FIRSTNAME : Robert

LASTNAME : Williams


ADDRESS :
45, Main Street

Reservation


[View Rooms](#)



FULL BOARD [DOUBLE ROOM]
ARRIVAL: 04/24/2022
DEPARTURE: 04/24/2022
Night(s): 1
PRICE: \$150



HALF BOARD [TRIPLE ROOM]
ARRIVAL: 04/24/2022
DEPARTURE: 04/24/2022
Night(s): 1
PRICE: \$168



FULL BOARD [TRIPLE ROOM]
ARRIVAL: 04/24/2022
DEPARTURE: 04/24/2022
Night(s): 1
PRICE: \$210

[← Back](#)

[Confirm →](#)

© Don Y.J Holdings

Figure 43: Interfaces - View Reservation Record

Edit Reservation Record:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(1\)](#) [Reports](#) [Admin](#) [Logout](#)

Update Reservation

Guest Details

FIRST NAME:

LAST NAME:

ADDRESS:

Reservation Information

Reservation Date:

Reservation ID:

Booking Status:

[Update](#)

© Don Y.J Holdings

Figure 44: Interfaces - Edit Reservation Record

View List of Administrators:

[Dashboard](#) [Rooms](#) [Room Type](#) [Reservations \(1\)](#) [Reports](#) [Admin](#) [Logout](#)

List of Admins

No.	<input type="checkbox"/> Name	Username	User Type	Mobile
<input type="checkbox"/>	Admin	admin	Administrator	755486440

[New](#) [Delete Selected](#)

© Don Y.J Holdings

Figure 45: Interfaces - View List of Administrators

Create New Administrator Account:

The screenshot shows a web application interface with a dark navigation bar at the top containing links: Dashboard, Rooms, Room Type, Reservations (1), Reports, Admin (highlighted), and Logout. Below the navigation bar is a light gray form titled "New Administrator Account". The form contains the following fields: "Name:" with a text input containing "Account Name"; "Username:" with a text input containing "Username"; "Password:" with a text input containing "Account Password"; "User Type:" with a dropdown menu set to "Administrator"; and "Mobile:" with a text input containing "Mobile Number". A blue "Save" button is located at the bottom of the form. Below the form, the text "© Don Y.J Holdings" is visible.

Figure 46: Interfaces - Create New Administrator Account

Update Administrator Account:

The screenshot shows the same web application interface as Figure 46, but with the "Admin" link highlighted in the navigation bar. Below the navigation bar is a light gray form titled "Update Account". The form contains the following fields: "Name:" with a text input containing "Admin"; "Username:" with a text input containing "admin"; "Password:" with a text input containing "Account Password"; "User Type:" with a dropdown menu set to "Administrator"; and "Mobile:" with a text input containing "755498440". A blue "Update" button is located at the bottom of the form. Below the form, the text "© Don Y.J Holdings" is visible.

Figure 47: Interfaces - Update Administrator Account

Room Booking Report Generation:

Dashboard
Rooms
Room Type
Reservations (1)
Reports
Admin
Logout

Room Booking Report
Date: 04/24/2022

Room
Status
Checkedin
Checkedout
Submit

Pending
Checkedin Date :2022-04-24 Checkedout Date :2022-04-24

Guest	Room	Price	Arrival	Departure	Night(s)	Subtotal
Robert Whilliams	TRIPLE ROOM [HALF BOARD]	\$ 168	04/24/2022	04/24/2022	1	\$ 168
Robert Whilliams	DOUBLE ROOM [FULL BOARD]	\$ 150	04/24/2022	04/24/2022	1	\$ 150
Robert Whilliams	TRIPLE ROOM [FULL BOARD]	\$ 210	04/24/2022	04/24/2022	1	\$ 210

Total Amount
Subtotal: \$ 528
Print

© Don Y.J Holdings

Figure 48: Interfaces - Room Booking Report Generation

Print Booking Report:

Holdings (P
4/24/2022, 4:47 PM

Don Y.J Holdings (Pvt) Ltd
Date: 04/24/2022

Pending
Checkedin Date :2022-04-24 Checkedout Date :2022-04-24

Guest	Room	Price	Arrival	Departure	Nights	Subtotal
Robert Whilliams	TRIPLE ROOM [HALF BOARD]	€ 168	04/24/2022	04/24/2022	1	€ 168
Robert Whilliams	DOUBLE ROOM [FULL BOARD]	€ 150	04/24/2022	04/24/2022	1	€ 150
Robert Whilliams	TRIPLE ROOM [FULL BOARD]	€ 210	04/24/2022	04/24/2022	1	€ 210

Total Amount
Total: € 528

Print
1 sheet of paper
Destination: Microsoft Print to PDF
Pages: All
Layout: Portrait
Color: Color
More settings
Print Cancel

Figure 49: Interfaces - Print Booking Report

Centralized Management System Login:

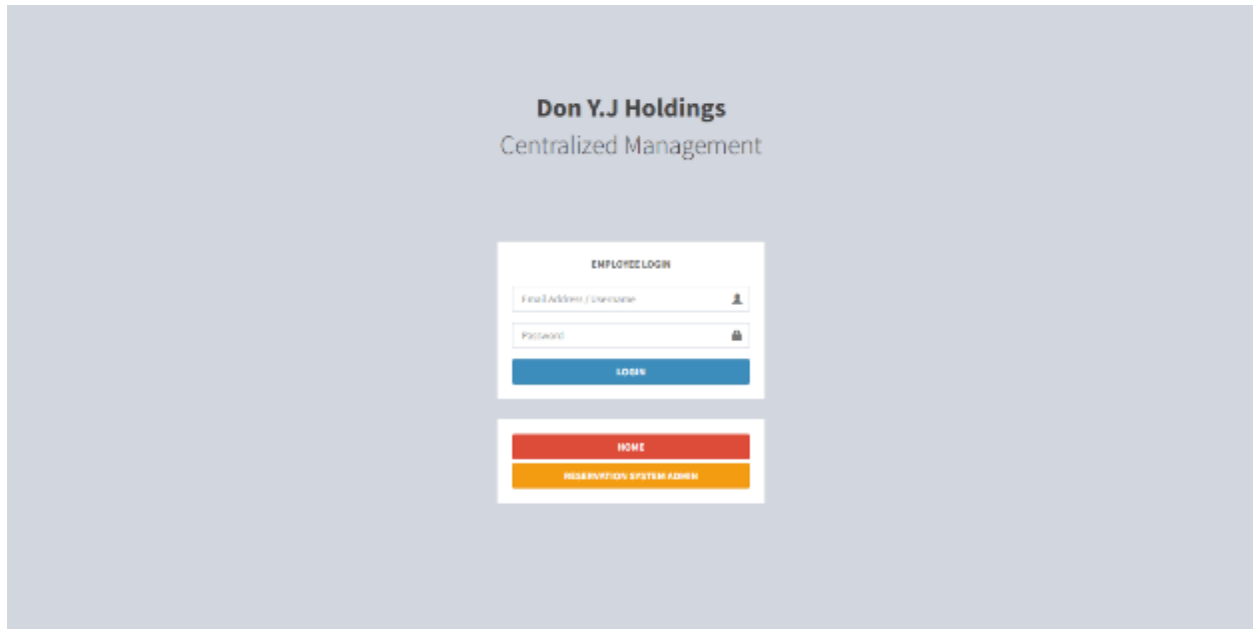


Figure 50: Interfaces - Centralized Management System Login

Management System Admin Dashboard:



Figure 51: Interfaces - Centralized Management Admin Dashboard

Create New Department:

The screenshot shows the 'Add Department' form within the Don Y.J Holdings system. The interface includes a top navigation bar with the company name and an 'Admin' link. A left sidebar contains a navigation menu with options like Home, Department, Staff Members, Staff Payroll, and Staff Leave. The main content area is titled 'Departments' and contains the 'Add Department' form. The form has a single input field labeled 'Department Name' and a green 'Submit' button. The footer of the page indicates the copyright year 2022.

Don Y.J Holdings

Administrator Available

Navigation: Home, Department, Staff Members, Staff Payroll, Staff Leave

Departments

Add Department

Department Name

Submit

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 52: Interfaces - Create New Department

View and Manage Existing Departments:

The screenshot shows the 'Manage Departments' table within the Don Y.J Holdings system. The interface includes a top navigation bar with the company name and an 'Admin' link. A left sidebar contains a navigation menu with options like Home, Department, Staff Members, Staff Payroll, and Staff Leave. The main content area is titled 'Departments' and contains the 'Manage Departments' table. The table has columns for '#', 'Department Name', and 'Actions'. The 'Actions' column contains 'Edit' and 'Delete' buttons for each department. The table lists 9 departments: Administration, Human Resources (HR), Security, Information Technology (IT), Sales and Marketing, Accounts and Finance, Food and Beverage, Housekeeping, and Front Office. The footer of the page indicates the copyright year 2022.

Don Y.J Holdings

Administrator Available

Navigation: Home, Department, Staff Members, Staff Payroll, Staff Leave

Departments

Manage Departments

Show 10 entries Search

#	Department Name	Actions
1	Administration	Edit Delete
2	Human Resources (HR)	Edit Delete
3	Security	Edit Delete
4	Information Technology (IT)	Edit Delete
5	Sales and Marketing	Edit Delete
6	Accounts and Finance	Edit Delete
7	Food and Beverage	Edit Delete
8	Housekeeping	Edit Delete
9	Front Office	Edit Delete

Showing 1 to 9 of 9 entries

Previous 1 Next

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 53: Interfaces - View and Manage Existing Departments

Add New Staff Member:

The screenshot shows the 'Add Staff' form within the 'Staff Management' section of the Don Y.J Holdings application. The form is titled 'Add Staff' and contains various input fields for staff details. The left sidebar shows the user is an Administrator. The top navigation bar includes 'Home', 'Staff Management', and 'Add Staff'.

Form Fields:

- Full Name:** Text input field.
- Department:** Dropdown menu.
- Gender:** Select dropdown.
- Email:** Text input field.
- Mobile:** Text input field.
- Photo:** 'Choose File' button, with 'No file chosen' text.
- Date of Birth:** Date picker (dd/mm/yyyy).
- Date of Joining:** Date picker (dd/mm/yyyy).
- City:** Text input field.
- State:** Text input field.
- Country:** Select dropdown.
- Password:** Text input field.
- Address:** Text area.

A green 'Submit' button is located at the bottom right of the form.

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 54: Interfaces - Add New Staff Member

View and Manage Existing Staff Records:

The screenshot shows the 'Manage Staff' table within the 'Staff Management' section of the Don Y.J Holdings application. The table displays a list of staff members with their details and actions. The left sidebar shows the user is an Administrator. The top navigation bar includes 'Home', 'Staff Management', and 'Manage Staff'.

Table Structure:

#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Jennifer Larry		Accounts and Finance	Female	2354187924	jennifer@outlook.com	04-06-1994	15-07-2020	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	Edit Delete
2	David Jones		Information Technology (IT)	Male	1254630879	david@gmail.com	14-07-1999	16-02-2019	56/67, Cross Street	Awissawella	Western	Sri Lanka	Edit Delete
3	Anne Statham		Human Resources (HR)	Female	4596320178	annastat@yahoo.com	12-11-1992	05-08-2020	789, Second Cross Street	Galle	Southern	Sri Lanka	Edit Delete
4	Robert Willeys		Administration	Male	7891230458	robertwills@gmail.com	09-11-1988	08-10-2015	56/98, First Street	Colombo 02	Western	Sri Lanka	Edit Delete

Showing 1 to 4 of 4 entries

Previous 1 Next

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 55: Interfaces - View and Manage Existing Staff Records

Edit Staff Member Record:

The screenshot shows the 'Edit Staff' form within the 'Staff Management' section of the 'Don Y.J Holdings' application. The interface includes a sidebar with navigation links: Home, Department, Staff Members, Staff Payroll, and Staff Leave. The main content area is titled 'Edit Staff' and contains various input fields for staff information. The fields are organized into two columns. The left column includes 'Full Name' (Jennifer Larry), 'Gender' (Select), 'Mobile' (2354187924), 'Date of Birth' (04/06/1994), 'City' (Trincmalee), and 'Country' (Sri Lanka). The right column includes 'Department' (Accounts and Finance), 'Email' (jennifer@outlook.com), 'Photo' (Choose File), 'Date of Joining' (15/07/2020), 'State' (Eastern), and 'Address' (7850E, Tens Apartment, Main Road). A green 'Submit' button is located at the bottom right of the form. The footer of the page indicates '© 2022 Don Y.J Holdings. All Rights Reserved.'

Full Name	Department
Jennifer Larry	Accounts and Finance

Gender	Email
Select	jennifer@outlook.com

Mobile	Photo
2354187924	Choose File No file chosen

Date of Birth	Date of Joining
04/06/1994	15/07/2020

City	State
Trincmalee	Eastern

Country	Address
Sri Lanka	7850E, Tens Apartment, Main Road

Submit

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 56: Interfaces - Edit Staff Member Record

Create Employee Payroll:

The screenshot shows the 'Add Employee Salary' form within the 'Salary' section of the 'Don Y.J Holdings' application. The interface includes a sidebar with navigation links: Home, Department, Staff Members, Staff Payroll, and Staff Leave. The main content area is titled 'Add Employee Salary' and contains a dropdown for 'Department Name' (Accounts and Finance) and a table for entering salary data. The table has four columns: 'Staff', 'Basic Salary (LKR)', 'Allowance (LKR)', and 'Total (LKR)'. The rows are for 'Jennifer Larry' and 'Roland Willis'. A green 'Submit' button is located at the bottom right of the form. The footer of the page indicates '© 2022 Don Y.J Holdings. All Rights Reserved.'

Staff	Basic Salary (LKR)	Allowance (LKR)	Total (LKR)
Jennifer Larry			
Roland Willis			

Submit

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 57: Interfaces - Create Employee Payroll

View and Manage Employee Payrolls:

Don Y.J Holdings | Administrator | Available

Salary Management | Home > Salary Management > Manage Salary

Manage Employee Payrolls

Show 10 entries

#	Staff Name	Department	Photo	Basic Salary	Allowance	Total Amount	Paid On	Invoice	Action
1	Roland Wills	Accounts and Finance		\$72000	\$6000	\$78000	2022-04-24	Invoice	Delete
2	Jennifer Larry	Accounts and Finance		\$80000	\$5000	\$85000	2022-04-24	Invoice	Delete
3	David Jones	Information Technology (IT)		\$97000	\$2550	\$99550	2022-04-24	Invoice	Delete
4	Robert Willeys	Administration		\$350000	\$25850	\$375850	2022-04-24	Invoice	Delete

Showing 1 to 4 of 4 entries

Previous 1 Next

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 58: Interfaces - View and Manage Employee Payrolls

Generate Employee Salary Invoice Generation:

Don Y.J Holdings | Administrator | Available

Salary Invoice #00237 | Home > Salary Management > Invoice

Don Y.J Holdings | Centralized Management System | Date: 24-04-2022

From: **Financial Controller**
Don Y.J Holdings (Pvt) Ltd
No. 17, Hewa Avenue,
Colombo 07, 00700
Sri Lanka
Telephone: 011 271 2713
Email: donyj@palmbeach.lk

To: **Robert Willeys**
Colombo 02
Western, Sri Lanka
Mobile: 7891230458
Email: robertwills@gmail.com

Invoice #009
Paid On: 24-04-2022

#	Basic Salary	Allowance	Subtotal
1	LKR 350000/=	LKR 25850/=	LKR 375850/=

Payment Method: **Sampath VISHWA** ONLINE BANKING

Salary Info

Subtotal:	LKR 375850/=
Tax (0%):	LKR 0/=
Total:	LKR 375850/=

Dear Robert Willeys, your employee salary has been transferred to your respective bank account. Your payment has been confirmed by the Financial Controller of Don Y.J Holdings, on 24-04-2022. Please contact the Financial Controller if you haven't received the payment. © Don Y.J Holdings.

[Print](#) [Generate PDF](#)

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 59: Interfaces - Generate Employee Salary Invoice

View and Manage Employee Leave Requests:

Don Y.J Holdings | Administrator | Available

Leave Management | Home > Leave Management > Approve Leave

Approve Leave

Show: 10 entries | Search:

#	Staff	Photo	Department	Reason	From	To	Description	Applied On	Actions
1	David Jones		Information Technology (IT)	Sick	27-04-2022	30-04-2022	Sick leave	2022-04-24	<button>Approve</button> <button>Reject</button>
2	Jennifer Larry		Accounts and Finance	Family gathering	25-04-2022	28-04-2022	Family gathering	2022-04-24	<button>Approve</button> <button>Reject</button>
3	Roland Wills		Accounts and Finance	Political appointment	04-05-2022	06-05-2022	Political appointment	2022-04-24	<button>Approve</button> <button>Reject</button>

Showing 1 to 3 of 3 entries | Previous 1 Next

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 60: Interfaces - View and Manage Employee Leave Requests

View Employee Leave History:

Don Y.J Holdings | Administrator | Available

Leave Management | Home > Leave Management > View Leave

View Leave

Show: 10 entries | Search:

#	Staff	Photo	Department	Reason	From	To	Status	Description	Applied On
1	Roland Wills		Accounts and Finance	Political appointment	04-05-2022	06-05-2022	Pending	Political appointment	2022-04-24
2	Jennifer Larry		Accounts and Finance	Family gathering	25-04-2022	28-04-2022	Pending	Family gathering	2022-04-24
3	David Jones		Information Technology (IT)	Sick	27-04-2022	30-04-2022	Pending	Sick leave	2022-04-24

Showing 1 to 3 of 3 entries | Previous 1 Next

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 61: Interfaces - View Employee Leave History

Management System Employee Dashboard:

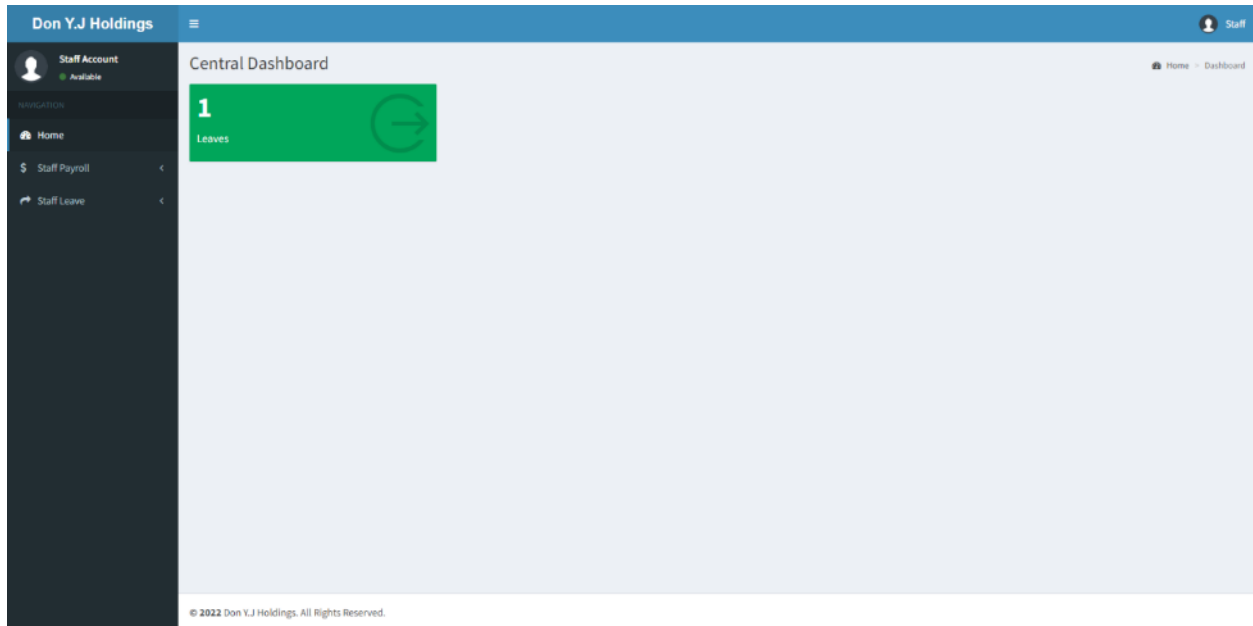


Figure 62: Interfaces - Management System Employee Dashboard

View and Print Salary Payment Invoices:

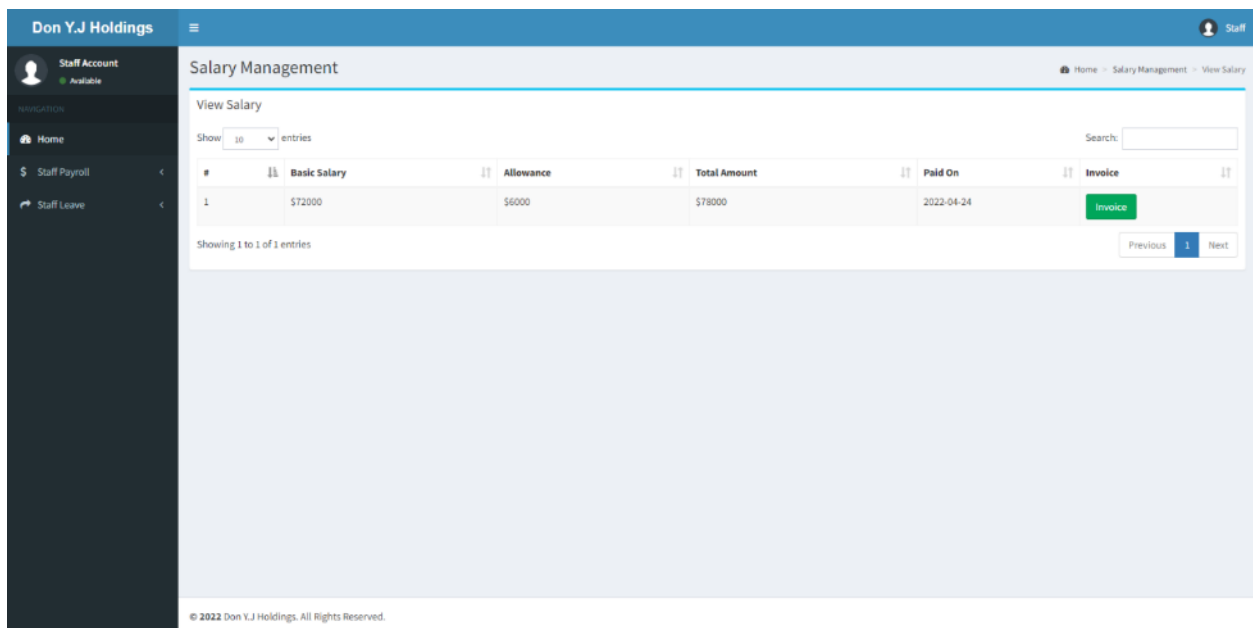


Figure 63: Interfaces - View and Print Salary Payment Invoices

Appy for Employee Leave:

The screenshot shows the 'Apply Leave' form within the 'Leave Management' section of the Don Y.J Holdings system. The form is titled 'Apply Leave' and contains several input fields: 'Reason' (a text box), 'Leave From' (a date picker showing 'dd/mm/yyyy'), 'Leave To' (a date picker showing 'dd/mm/yyyy'), and 'Description' (a text box). A green 'Submit' button is located at the bottom right of the form. The left sidebar shows the user's profile as 'Staff Account' with a status of 'Available'. The top navigation bar includes 'Home', 'Staff Payroll', and 'Staff Leave'. The footer indicates '© 2022 Don Y.J Holdings. All Rights Reserved.'

Figure 64: Interfaces - Appy for Employee Leave

View Leave Status History:

The screenshot shows the 'View Leave Status History' table within the 'Leave Management' section of the Don Y.J Holdings system. The table is titled 'View Leave' and displays a list of leave entries. The table has columns for '#', 'Reason', 'From', 'To', 'Status', 'Description', and 'Applied On'. A single entry is shown with the status 'Pending'. The table includes a search bar and pagination controls at the bottom.

#	Reason	From	To	Status	Description	Applied On
1	Political appointment	04-05-2022	06-05-2022	Pending	Political appointment	24-04-2022

Showing 1 to 1 of 1 entries

Previous 1 Next

Figure 65: Interfaces - View Leave Status History

Database Structure:

MySQL has been chosen as the preferred database system technology for the current application development. MySQL is an open-source and relational database system, developed by Oracle. The database structure of current application could be categorized into two sections.

Below shown is the database structure of the online reservation system.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> roomType	★ Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> auto	★ Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> firstSection	★ Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> guest	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> dining	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> payment	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	48.0 KiB	-
<input type="checkbox"/> reservation	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	64.0 KiB	-
<input type="checkbox"/> room	★ Browse Structure Search Insert Empty Drop	12	InnoDB	latin1_swedish_ci	32.0 KiB	-
<input type="checkbox"/> roomAccommodation	★ Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> gallery	★ Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> system	★ Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> user	★ Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> settings	★ Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16.0 KiB	-
13 tables	Sum	35	InnoDB	utf8mb4_general_ci	304.0 KiB	0 B

☐ Check all With selected: ▾

Figure 66: Database Structure - Online reservation system

Below shown is the database structure of the centralized management system.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> countryList	★ Browse Structure Search Insert Empty Drop	246	MyISAM	utf8_general_ci	10.9 KiB	-
<input type="checkbox"/> department	★ Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> leaveRequest	★ Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> userLogin	★ Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> payroll	★ Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> employee	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
6 tables	Sum	285	InnoDB	utf8mb4_general_ci	90.9 KiB	0 B

☐ Check all With selected: ▾

Figure 67: Database Structure - Centralized management system

Code Implementation:

Development would be the most crucial phase of the project, which also takes up the most timeframe of the entire project workplan. The system code needs to be properly implemented, in order for the project's online reservation and centralized management system to be functional.

For example, considering the web application code, the user system login page would be considered as the most essential aspect of the system. The application currently contains two types of user login, separated as guest login and employee login. Both users would require a unique username linked with a password in order to access the system.

Guest Login:

The guest login would be interacted by the company's guests requiring to reserve rooms. The user would be required to create an account by providing personal information, username and a password. The guest can then proceed to login to the system using the username and password.

The guest login screen would look as the snippet shown below. The user would need to enter his/her unique username and password, and then click the "Login" to access the system, which would redirect the system to the application's homepage and allow further functionalities.

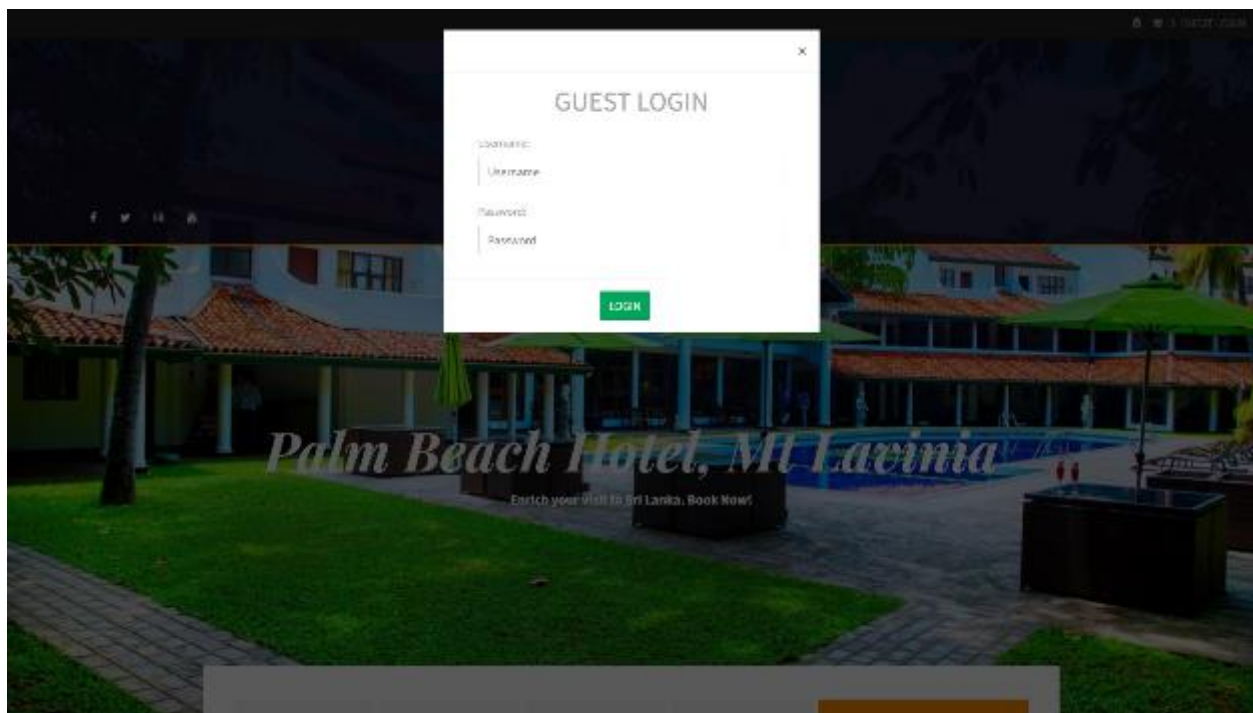


Figure 68: Guest login - user interface

In continuation of the example, the code snippet for the guest login interface is shown below.

```
logininfo.php X
booking > logininfo.php
1  <?php
2
3  if (!isset($_SESSION['dragonhouse_cart'])) {
4      # code...
5      redirect(WEB_ROOT.'index.php');
6  }
7
8
9
10
11  <div class="p-4"></div>
12
13  <h1 class="text-center" style="display: inline-block;">Login</h1>
14  <a style="display: inline-block;" href="personalinfo.php" data-title="Register New Guest" data-toggle="lightbox"> Register</a>
15
16  <div class="p-4"></div>
17
18  <form action="<?php echo WEB_ROOT.'login.php' ?>" method="post">
19      <div class="form-group">
20          <div class="">
21              <label class="control-label" for=
22                  "Username">Username:</label>
23              <input id="username" name="username" placeholder="Username" type="text" class="form-control input" style="width: 100%">
24          </div>
25
26          <br>
27
28          <div class="">
29              <label class="control-label" for=
30                  "pass">Password:</label>
31              <input name="pass" id="pass" placeholder="Password" type="password" class="form-control input" style="width: 100%">
32          </div>
33      </div>
34
35      <br>
36
37      <button type="submit" name="gsubmit" class="button">Sign In</button>
38  </form>
39
40  <div class="p-4"></div>
41
42
43
44  <br>
45
46
```

Figure 69: Guest login interface - code snippet

The guest would only be allowed in the system if the entered username and password credentials are validated by the application. The system uses PHP scripts in order to validate if the user credentials are correct, with a corresponding validation with the application database.

The PHP code snippet used to validate and verify the login function is shown below.

```
login.php
login.php
1  <?php
2  require_once ("includes/initialize.php");
3
4  if(isset($_POST['gsubmit'])){
5
6      $email = trim($_POST['username']);
7      $upass  = trim($_POST['pass']);
8      $h_upass = sha1($upass);
9
10     if ($email == '' OR $upass == '') {
11         message("Invalid Username or Password!", "error");
12         redirect(WEB_ROOT."booking/index.php?view=logininfo");
13     } else {
14         $guest = new Guest();
15         $res = $guest::guest_login($email,$h_upass);
16
17         if ($res==true){
18             redirect(WEB_ROOT."booking/index.php?view=payment");
19         }else{
20             message("Invalid Username and Password! Please contact administrator", "error");
21             redirect(WEB_ROOT."booking/index.php?view=logininfo");
22         }
23     }
24 }
25 }
26 }
27 ?>
28
```

Figure 70: Guest login verification - code snippet

Guest Registration:

If the guest doesn't already have an account, the user can click the "New User?" button, to create a new account after providing the required details, like the interface shown below.

Once the guest fills up the required information to register in the system, the data would then be stored in the system database. A sample filled user registration form would look like below;

The screenshot displays a web interface for guest registration. On the left, there is a 'Login' section with fields for 'Username' and 'Password', and a 'Sign in' button. The main part of the page is a 'Register New Guest' form. The form includes the following fields and values:

- FIRST NAME: John
- LAST NAME: Howard
- CITY: Colombo
- ADDRESS: 56/87, Greens Avenue, Main Street, Colombo 07
- DATE OF BIRTH: 05/02/2001
- MOBILE: 0789510237
- NATIONALITY: Sri Lankan
- COMPANY: (empty)
- COMPANY ADDRESS: (empty)
- EMAIL: johnhoward@gmail.com
- USERNAME: johnhoward123
- PASSWORD: (masked with asterisks)
- POSTAL CODE: 00700

At the bottom of the form, there is a checkbox for 'I Agree to the TERMS AND CONDITIONS of Don V.J. Hotel' and a 'Confirm' button. The background of the page features a hotel lobby with a 'Palm' sign and a swimming pool area with a 'RESERVE NOW' button.

Figure 71: Guest user registration - user interface

The code snippet for the user registration interface is shown below.

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 | <?php include '../layouts/head.php' ?>
6 </head>
7
8 <body id="page-top">
9 | <div id="wrapper">
10
11 | <?php include '../layouts/sidebar.php' ?>
12
13 | <div id="content-wrapper" class="d-flex flex-column">
14 | | <div id="content">
15 | |
16 | | <?php include '../layouts/topbar.php' ?>
17 | |
18 | | <!-- Container Fluid -->
19 | | <div class="container-fluid" id="container-wrapper">
20 | | | <div class="d-sm-flex align-items-center justify-content-between mb-4">
21 | | | | <h1 class="h3 mb-0 text-gray-800">User Registration</h1>
22 | | | </div>
23 | | |
24 | | | <div class="row">
25 | | | |
26 | | | | <div class="col-12">
27 | | | | | <!-- Form -->
28 | | | | | <div class="card">
29 | | | | | | <div class="card-header py-3 d-flex flex-row align-items center justify-content-between">
30 | | | | | | | <h6 class="m-0 font-weight-bold text-primary">New User</h6>
31 | | | | | | </div>
32 | | | | | |
33 | | | | | | <div class="card-body">
34 | | | | | | | <form action="" method="POST">
35 | | | | | | | |
36 | | | | | | | | <div class="row">
37 | | | | | | | | |
38 | | | | | | | | | <div class="col-12 col-md-6">
39 | | | | | | | | | | <div class="form-group">
40 | | | | | | | | | | | <label>Name</label>
41 | | | | | | | | | | | <input type="text" class="form-control" placeholder="Enter Name" name="name" required>
42 | | | | | | | | | | </div>
43 | | | | | | | | | </div>
44 | | | | | | | | |
45 | | | | | | | | | <div class="col-12 col-md-6">
46 | | | | | | | | | | <div class="form-group">
47 | | | | | | | | | | | <label>Date of Birth</label>
48 | | | | | | | | | | | <input type="date" class="form-control" placeholder="Enter Date of Birth" name="dob" required>
49 | | | | | | | | | | </div>
50 | | | | | | | | | </div>
51 | | | | | | | | |
52 | | | | | | | | | <div class="col-12">
53 | | | | | | | | | | <div class="form-group">
54 | | | | | | | | | | | <label>Address</label>
55 | | | | | | | | | | | <input type="text" class="form-control" placeholder="Enter Address" name="address" required>
56 | | | | | | | | | | </div>
57 | | | | | | | | | </div>
58 | | | | | | | | |
59 | | | | | | | | | <div class="col-12 col-md-6">
60 | | | | | | | | | | <div class="form-group">
61 | | | | | | | | | | | <label>Email Address</label>
62 | | | | | | | | | | | <input type="text" class="form-control" placeholder="Enter Email Address" name="email" required>
63 | | | | | | | | | | </div>
64 | | | | | | | | | </div>
65 | | | | | | | | |
66 | | | | | | | | | </div>
67 | | | | | | | | </div>
68 | | | | | | | </div>
69 | | | | | | </div>
70 | | </div>
71 | </div>
72 </body>
73 </html>
```

Figure 72: Guest user registration - code snippet #1

```

68         <div class="col-12 col-md-6">
69             <div class="form-group">
70                 <label>Mobile</label>
71                 <input type="text" class="form-control" placeholder="Enter Mobile Number" name="mobile" required>
72             </div>
73         </div>
74
75         <div class="col-12 col-md-6">
76             <div class="form-group">
77                 <label>NTC</label>
78                 <input type="text" class="form-control" placeholder="Enter NTC Number" name="ntc" required>
79             </div>
80         </div>
81
82         <div class="col-12 col-md-6">
83             <div class="form-group">
84                 <label>Gender</label>
85                 <select name="gender" class="form-control">
86                     <option value="Male">Pick Gender</option>
87                     <option value="Female">Male</option>
88                     <option value="Female">Female</option>
89                 </select>
90             </div>
91         </div>
92
93         <div class="col-12 col-md-6">
94             <div class="form-group">
95                 <label>Username</label>
96                 <input type="text" class="form-control" placeholder="Enter Username" name="username" required>
97             </div>
98         </div>
99
100        <div class="col-12 col-md-6">
101            <div class="form-group">
102                <label>Password</label>
103                <input type="password" class="form-control" placeholder="Enter Password" name="password" required>
104            </div>
105        </div>
106
107    </div>
108
109    <div class="form-group">
110        <input type="submit" class="btn btn-block btn-outline-primary" name="send">
111    </div>
112
113    </form>
114
115 </div>
116
117 </div>
118
119 </div>
120
121
122 </div>
123 <!-- Row -->
124
125 <?php include '../layouts/logout.php' ?>
126
127 </div>
128 <!-- Container Fluid -->
129 </div>
130
131 <?php include '../layouts/footer.php' ?>
132
133 </div>
134
135 </div>
136
137 <?php include '../layouts/scroll.php' ?>
138
139 <?php include '../layouts/script.php' ?>
140
141 </body>
142
143 </html>

```

Figure 73: Guest user registration - code snippet #2

Once the guests have entered the required information to create an account, the user information would be stored in the system database. The application uses PHP scripts in order to validate the provided information and save them in the relevant database system.

The PHP code snippet used to validate and save user registrations is shown below.

```
143 <?php
144
145 if (isset($_POST['send'])) {
146
147     $name=$_POST['name'];
148     $dob=$_POST['dob'];
149     $address=$_POST['address'];
150     $gender=$_POST['gender'];
151     $email=$_POST['email'];
152     $mobile=$_POST['mobile'];
153     $nic=$_POST['nic'];
154     $regDate=$_POST['date'];
155     $userName=$_POST['userName'];
156     $password=$_POST['password'];
157     $userType=$_POST['userType'];
158
159     $con=mysqli_connect("localhost","root","","reserve");
160
161     $sql="INSERT INTO user (name,dob,address,gender,email,mobile,nic,regDate,username,password,utypeid)
162     VALUES ('$name','$dob','$address','$gender','$email','$mobile','$nic','$regDate','$userName','$password','$userType')";
163
164     $query=mysqli_query($con,$sql);
165
166     if ($query) {
167         echo '<script>alert("Registration Successful!")</script>';
168     }
169     else{
170         echo '<script>alert("Registration Failed! Please re-check provided information")</script>';
171     }
172
173 }
174
175 ?>
```

Figure 74: User registration validation - code snippet

Employee Login:

If the company employees are required to login to the system, they would already receive a username and a password which would be created by an administrative member of the company, including the general manager. The company administrative team and the general manager would have the ability to create new employee records. In order to create a new employee record, the application would require personal information of the specific employee. Following, a unique username and password would be provided to login in the system and access further functions.

The employee login screen would look as the snippet shown below. The employee would need to enter his/her unique username and password, and then click the “Login” to access the system, which would redirect the system to the employee dashboard and allow further functionalities.

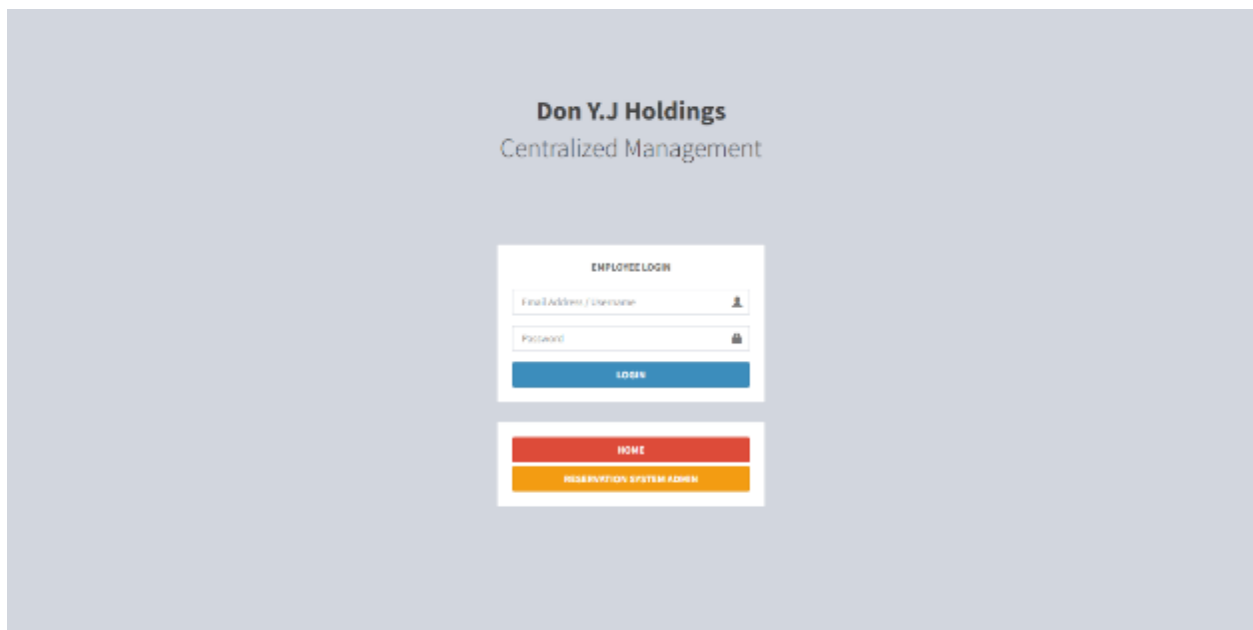
The image shows a user interface for an employee login system. At the top, the text "Don Y.J Holdings" is displayed in a bold, dark font, with "Centralized Management" in a smaller, lighter font below it. The main content area is a light gray rectangle. Inside this rectangle, there is a white box titled "EMPLOYEE LOGIN". This box contains two input fields: the first is labeled "Email Address / Username" and has a user icon to its right; the second is labeled "Password" and has a lock icon to its right. Below these fields is a blue button labeled "LOGIN". Below the white box, there are two more buttons: a red one labeled "HOME" and an orange one labeled "RESERVATION SYSTEM ADMIN".

Figure 75: Employee login - user interface

Once the employee enters the login credentials, the credentials would be verified by the system, if they already exist in the system. If the entered credentials are correctly validated by the system, it would redirect to the employee dashboard page, where the employee functionalities would be available. However, employees would not be able to create their own accounts, in consideration to system security and company data privacy.

In continuation of the example, the code snippet for employee login interface is shown below.

```

28 <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600,700,300italic,400italic,600italic">
29 </head>
30 <body class="hold-transition login-page">
31 <div class="login-box">
32 <div class="login-logo">
33 <a href="#"><b>Don Y.J Holdings</b></a>
34 <a href="#">Centralized Management</a>
35 </div>
36 <br><br><br><br>
37 <!-- /.login-logo -->
38 <div class="login-box-body">
39 <p class="login-box-msg"><b>EMPLOYEE LOGIN</b></p>
40
41 <?php echo form_open('Home/login'); ?>
42 <div class="form-group has-feedback">
43 <input type="text" name="txtusername" class="form-control" placeholder="Email Address / Username">
44 <span class="glyphicon glyphicon-user form-control-feedback"></span>
45 </div>
46 <div class="form-group has-feedback">
47 <input type="password" name="txtpassword" class="form-control" placeholder="Password">
48 <span class="glyphicon glyphicon-lock form-control-feedback"></span>
49 </div>
50 <?php echo $this->session->flashdata('login_error'); ?>
51 <div class="row">
52 <!-- /.col -->
53 <div class="col-xs-12 pull-left">
54 <button type="submit" class="btn btn-primary btn-block"><b>LOGIN</b></button>
55 </div>
56 <!-- /.col -->
57 </div>
58 </form>
59 </div>
60
61 <div class="row">
62 <div class="col-12">
63 <div class="login-box">
64 <div class="login-box-body">
65 <fieldset>
66 <a class="btn btn-danger btn-block" href="../../donhotel"><b>HOME</b></a>
67 <a class="btn btn-warning btn-block" href="../../donhotel/admin"><b>RESERVATION SYSTEM ADMIN</b></a>
68 </fieldset>
69 </div>
70 </div>
71 </div>
72 </div>
73
74 <!-- /.login-box-body -->
75 </div>
76 <!-- /.login-box -->
77
78 <!-- jQuery 3 -->
79 <script src="<?php echo base_url(); ?>assets/bower_components/jquery/dist/jquery.min.js"></script>
80 <!-- Bootstrap 3.3.7 -->
81 <script src="<?php echo base_url(); ?>assets/bower_components/bootstrap/dist/js/bootstrap.min.js"></script>
82 <!-- iCheck -->
83 <script src="<?php echo base_url(); ?>assets/plugins/iCheck/iCheck.min.js"></script>
84 <script>
85 $(function () {
86 $('input').iCheck({
87 checkboxClass: 'i-checkbox square-blue',

```

Figure 76: Employee login interface - code snippet

The employee would only be allowed in the system if the entered username and password credentials are validated by the application. The system uses PHP scripts in order to validate if the user credentials are correct, with a corresponding validation with the application database.

The PHP code snippet used to validate and verify the login function is shown below.

```
45 <?php
46
47 if (isset($_POST['send'])) {
48
49     $username=$_POST['username'];
50     $password=$_POST['password'];
51
52     $con=mysqli_connect("localhost","root","","reserve");
53
54     if ($con) {
55         echo "Connection Error!";
56     }
57
58     $sql="SELECT * FROM user u
59     INNER JOIN usertype ut ON u.utypeid=ut.utypeid
60     WHERE u.username = '$username' AND u.password = '$password'";
61
62     $query=mysqli_query($con,$sql);
63
64     $count=mysqli_num_rows($query);
65
66     if ($count<1) {
67         ?>
68         <script>alert("Incorrect Username or Password")</script>
69         <?php
70     }
71     else {
72
73         while ($row=mysqli_fetch_array($query)) {
74
75             $_SESSION['loginName']=$row['name'];
76             $_SESSION['loginutype']=$row['utype'];
77             $_SESSION['loginid']=$row['uid'];
78
79         }
80
81         ?>
82
83         <script>window.location.href="pages/index.php"</script>
84
85         <?php
86     }
87
88 }
89
90 ?>
91
```

Figure 77: Employee login verification - code snippet

Employee Registration:

An administrative staff member of the company would be able to create employee records after providing their personal information in the system, like the interface shown below;

The screenshot displays the 'Don Y.J Holdings' web application. The top navigation bar includes the company name and an 'Admin' link. A left sidebar shows the user profile as 'Administrator' and a navigation menu with options: Home, Department, Staff Members, Staff Payroll, and Staff Leave. The main content area is titled 'Staff Management' and contains an 'Add Staff' form. The form fields are organized into two columns: Full Name, Gender, Mobile, Date of Birth, City, Country, and Address on the left; and Department, Email, Photo, Date of Joining, State, and Password on the right. A 'Submit' button is located at the bottom right of the form. The footer indicates '© 2022 Don Y.J Holdings. All Rights Reserved.'

Figure 78: Employee registration - user interface

Once the staff fills up the required information to register in the system, the data would then be stored in the system database. A sample filled user registration form would look like below;

This screenshot shows the same 'Add Staff' form as Figure 78, but with sample data entered. The 'Full Name' field contains 'Taylor James', 'Gender' is set to 'Select', 'Mobile' is '941237951', 'Date of Birth' is '08/11/1999', 'City' is 'Dehiwala', 'Country' is 'Sri Lanka', and 'Address' is 'No. 56, Central Avenue, Dehiwala'. On the right side, 'Department' is 'Front Office', 'Email' is 'jamestaylor34@yahoo.com', 'Photo' shows a file named 'avatar7.1ee966d656f0.png', 'Date of Joining' is '08/02/2022', 'State' is 'Western', and 'Password' is masked with asterisks. The 'Submit' button remains at the bottom right.

Figure 79: Sample employee registration - filled form

The code snippet for the employee registration interface is shown below.

```
add-staff.php X
application > views > admin > add-staff.php
1 <!-- Content Wrapper. Contains page content -->
2 <div class="content-wrapper">
3 <!-- Content Header (Page header) -->
4 <section class="content-header">
5 <h1>
6 Staff Management
7 </h1>
8 <ol class="breadcrumb">
9 <li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>
10 <li><a href="#">Staff Management</a></li>
11 <li class="active">Add Staff</li>
12 </ol>
13 </section>
14
15 <!-- Main content -->
16 <section class="content">
17 <div class="row">
18
19 <?php echo validation_errors('<div class="col-md-12">
20 <div class="alert alert-danger alert-dismissible">
21 <button type="button" class="close" data-dismiss="alert" aria-hidden="true"></button>
22 <h4><i class="icon fa fa-check"></i> Failed!</h4>','</div>
23 </div>'); ?>
24
25 <?php if($this->session->flashdata('success')): ?>
26 <div class="col-md-12">
27 <div class="alert alert-success alert-dismissible">
28 <button type="button" class="close" data-dismiss="alert" aria-hidden="true"></button>
29 <h4><i class="icon fa fa-check"></i> Success!</h4>
30 <?php echo $this->session->flashdata('success'); ?>
31 </div>
32 </div>
33 <?php elseif($this->session->flashdata('error')): ?>
34 <div class="col-md-12">
35 <div class="alert alert-danger alert-dismissible">
36 <button type="button" class="close" data-dismiss="alert" aria-hidden="true"></button>
37 <h4><i class="icon fa fa-check"></i> Failed!</h4>
38 <?php echo $this->session->flashdata('error'); ?>
39 </div>
40 </div>
41 <?php endif; ?>
42
43 <!-- column -->
44 <div class="col-md-12">
45 <!-- general form elements -->
46 <div class="box box-info">
47 <div class="box-header with-border">
48 <h3 class="box-title">Add Staff</h3>
49 </div>
50 <!-- /.box-header -->
51 <!-- form start -->
52 <?php echo form_open_multipart('Staff/insert'); ?>
53 <div class="box-body">
54 <div class="col-md-6">
55 <div class="form-group">
56 <label>Full Name</label>
57 <input type="text" name="txname" class="form-control" placeholder="Full Name">
58 </div>
59 </div>
60
61 <div class="col-md-6">
62 <div class="form-group">
63 <label>Department</label>
64 <select class="form-control" name="slcdepartment">
65 <option value="">Select</option>
66 <?php
67 if(isset($department))
68 {
69 foreach($department as $cnt)
70 {
71 print "<option value='".$cnt['id']."'>".$cnt['department_name']."</option>";
72 }
73 }
74 </select>
75 </div>
76 </div>
77 </div>
```

Figure 80: Employee registration - code snippet #1

```

91     <div class="col-md-6">
92         <div class="form-group">
93             <label>Email</label>
94             <input type="text" name="txtemail" class="form-control" placeholder="Email">
95         </div>
96     </div>
97
98     <div class="col-md-6">
99         <div class="form-group">
100             <label>Mobile</label>
101             <input type="text" name="txtmobile" class="form-control" placeholder="Mobile">
102         </div>
103     </div>
104
105     <div class="col-md-6">
106         <div class="form-group">
107             <label>Photo</label>
108             <input type="file" name="filephoto" class="form-control">
109         </div>
110     </div>
111
112     <div class="col-md-6">
113         <div class="form-group">
114             <label>Date of Birth</label>
115             <input type="date" name="txtdob" class="form-control" placeholder="DOB">
116         </div>
117     </div>
118
119     <div class="col-md-6">
120         <div class="form-group">
121             <label>Date of Joining</label>
122             <input type="date" name="txtdoj" class="form-control" placeholder="DOJ">
123         </div>
124     </div>
125
126     <div class="col-md-6">
127         <div class="form-group">
128             <label>City</label>
129             <input type="text" name="txtcity" class="form-control" placeholder="City">
130         </div>
131     </div>
132
133     <div class="col-md-6">
134         <div class="form-group">
135             <label>State</label>
136             <input type="text" name="txtstate" class="form-control" placeholder="State">
137         </div>
138     </div>
139
140     <div class="col-md-6">
141         <div class="form-group">
142             <label>Country</label>
143             <select class="form-control" name="slcountry">
144                 <option value="">Select</option>
145                 <?php
146                     if(isset($country))
147                     {
148                         foreach ($country as $cnt1)
149                         {
150                             print "<option value='". $cnt1['country_name']."'>". $cnt1['country_name']."'</option>";
151                         }
152                     }
153                 <?>
154             </select>
155         </div>
156     </div>
157
158     <div class="col-md-6">
159         <div class="form-group">
160             <label>Password</label>
161             <input type="password" name="password" class="form-control" placeholder="Password">
162         </div>
163     </div>
164
165     <div class="col-md-12">
166         <div class="form-group">
167             <label>Address</label>
168             <textarea class="form-control" name="txtaddress" placeholder="Address"></textarea>
169         </div>
170     </div>
171
172 </div>
173 <!-- /.box-body -->
174 <div class="box-footer">
175     <button type="submit" class="btn btn-success pull-right">Submit</button>

```

Figure 81: Employee registration - code snippet #2

Once the administration staff have entered the required information to register an employee, the employee information would be stored in the system database. The application uses PHP scripts in order to validate the provided information and save them in the relevant database system.

The PHP code snippet used to validate and save employee registrations is shown below.

```

32 public function insert()
33 {
34     $this->form_validation->set_rules('txtname', 'Full Name', 'required');
35     $this->form_validation->set_rules('slcgender', 'Gender', 'required');
36     $this->form_validation->set_rules('slcdepartment', 'Department', 'required');
37     $this->form_validation->set_rules('txtemail', 'Email', 'trim|required|valid_email');
38     $this->form_validation->set_rules('txtmobile', 'Mobile Number', 'required|regex_match[/^[0-9]{10}$/]/');
39     $this->form_validation->set_rules('txtdob', 'Date of Birth', 'required');
40     $this->form_validation->set_rules('txt doj', 'Date of Joining', 'required');
41     $this->form_validation->set_rules('txtcity', 'City', 'required');
42     $this->form_validation->set_rules('txtstate', 'State', 'required');
43     $this->form_validation->set_rules('slccountry', 'Country', 'required');
44     $this->form_validation->set_rules('password', 'Password', 'required');
45
46     $name=$this->input->post('txtname');
47     $gender=$this->input->post('slcgender');
48     $department=$this->input->post('slcdepartment');
49     $email=$this->input->post('txtemail');
50     $mobile=$this->input->post('txtmobile');
51     $dob=$this->input->post('txtdob');
52     $doj=$this->input->post('txt doj');
53     $city=$this->input->post('txtcity');
54     $state=$this->input->post('txtstate');
55     $country=$this->input->post('slccountry');
56     $pass=$this->input->post('password');
57     $address=$this->input->post('txtaddress');
58     $added=$this->session->userdata('userid');
59
60     if($this->form_validation->run() != false)
61     {
62         $this->load->library('image_lib');
63         $config['upload_path'] = 'uploads/profile-pic/';
64         $config['allowed_types'] = 'gif|jpg|png|jpeg';
65         $this->load->library('upload', $config);
66         if ( ! $this->upload->do_upload('filephoto'))
67         {
68             $image='default-pic.jpg';
69         }
70         else
71         {
72             $image_data = $this->upload->data();
73
74             $config = array(
75                 'image_library' => 'gd2',
76                 'source_image' => $image_data['full_path'],
77                 'maintain_ratio' => TRUE,
78                 'width' => 150,
79                 'height' => 150,
80                 'quality' => 50
81             );
82             $this->image_lib->clear();
83             $this->image_lib->initialize($config);
84             $this->image_lib->resize();
85
86             $image=$image_data['file_name'];
87         }
88         $login=$this->Home_model->insert_login(array('username'=>$email,'password'=>$pass,'usertype'=>2));
89         if($login>0)
90         {
91             $data=$this->Staff_model->insert_staff(array('id'=>$login,'staff_name'=>$name,'gender'=>$gender,'email'=>$email,'mobile'=>$mobile,'dob
92             '=>$dob,'doj'=>$doj,'city'=>$city,'state'=>$state,'country'=>$country,'password'=>$pass,'address'=>$address));
93
94             if($data==true)
95             {
96                 $this->session->set_flashdata('success', "New Staff Added Successfully");
97             }else{
98                 $this->session->set_flashdata('error', "Sorry, New Staff Adding Failed.");
99             }
100         }
101         redirect($ _SERVER['HTTP_REFERER']);

```

Figure 82: Employee registration validation - code snippet

Session:

Session implementation is essential throughout the application in order to acknowledge user roles in the system. Based on the system requirements, only the company employees should be able to have high level access to company information. Guests should only be able to view public company information as well as specific reservation-related information.

To achieve this, a special PHP variable “session” has been implemented throughout the application to allow special user permission to selected user types. This variable assists the application to separate user functionalities based on their unique user types.

The snippet below shows how PHP session is implemented during every login session.

```
1  <?php
2  session_start();
3  ?>
4
5  <?php
6  $loginid=0;
7  $loginid=$_SESSION['loginid'];
8
9  if ($loginid==0) {
10     ?>
11     <script>
12         window.location.href="http://localhost/reservation_system/";
13     </script>
14     <?php
15     }
16 ?>
17
18 <?php
19 $url="http://localhost/reservation_system/"
20 ?>
```

Figure 83: Session implementation during user login - code snippet

Similarly, the snippet below shows how the same session would be destroyed after any user logout from the snippet, which erases any user functionality related with the specific user type.

```
1  <?php
2
3      session_start();
4      session_destroy();
5
6  ?>
7
8  <script>window.location.href="../index.php"</script>
```

Figure 84: Session implementation during user logout - code snippet

Development Tools Utilization:

Sublime Text:

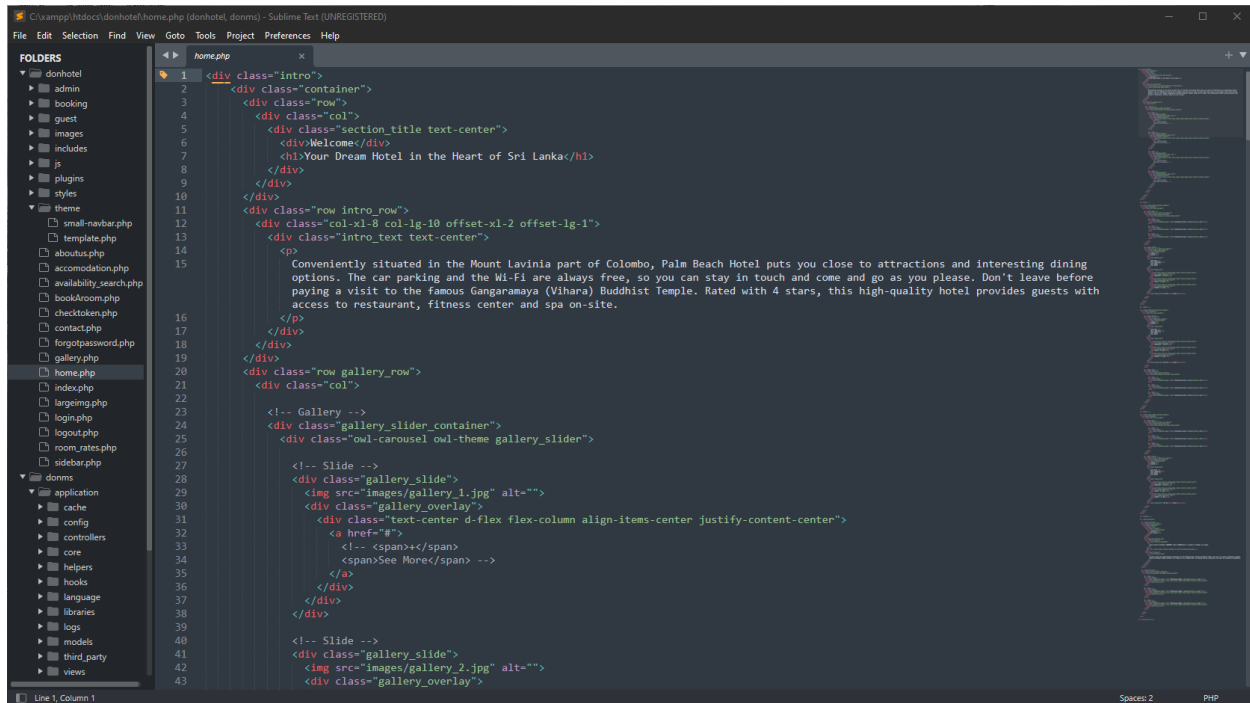


Figure 85: Development Tools Utilization - Sublime Text

XAMPP Server:

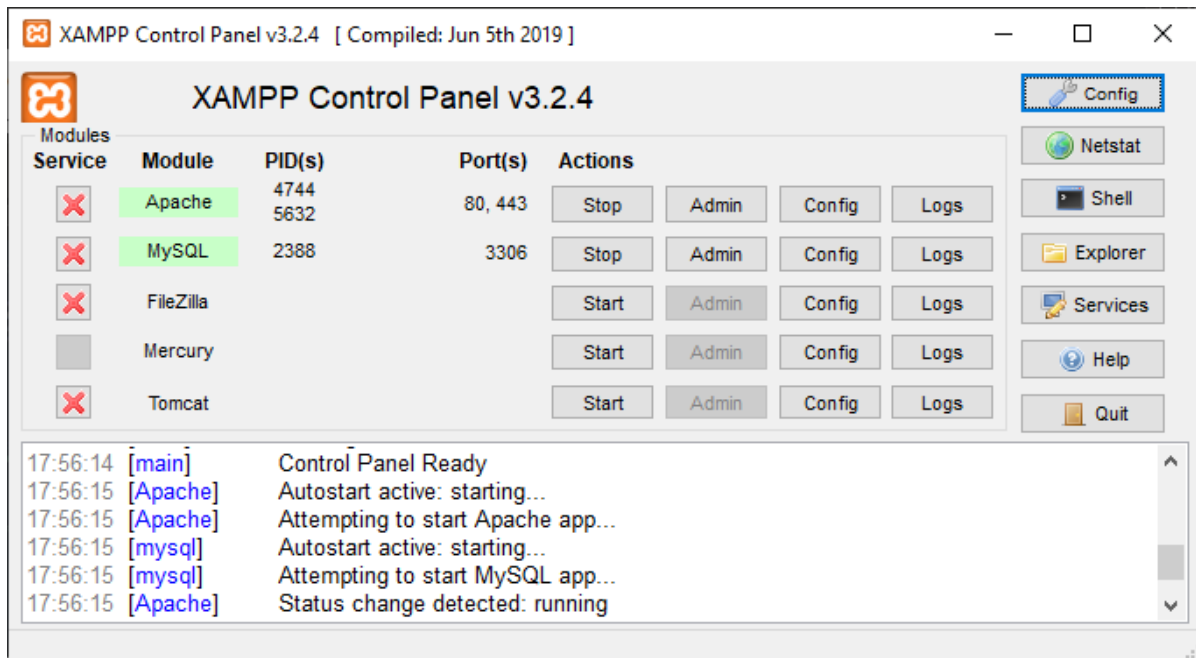


Figure 86: Development Tools Utilization - XAMPP Server

5. Validation

Validation would be considered as an important section of any software project. The validation factor of the project ensures that the implementation of the project corresponds with the initial requirements and analysis. Validation would be advantageous in the current project to confirm that the project flow is moving towards the proper direction in consideration of the requirements gathered using proper research and system analysis. This process helps the project planners understand if the necessary data are being collected and evaluated for the project.

Software validation is technically a process of analysing and assessing the specific application, along with the initial software planning phase, to make sure that the final product would actually meet the proper requirements, based on the pre-defined plans. It is essential to confirm that the development actually benefits businesses and customers in the real world, after software deployment. This also ensures that the proper business and customer requirements are properly considered during the implementation stage (ProfessionalQA, 2020).

Software validations and verification are technically conducted in a proper flow during every project. The illustration below shows a proper flow that fits similar projects;

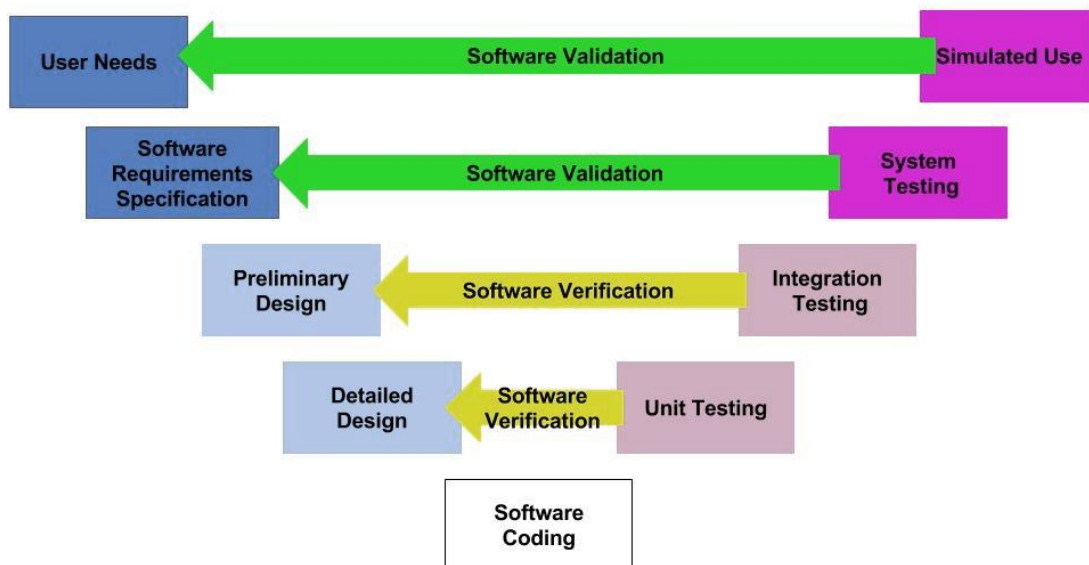


Figure 87: Software validation and verification

Online Reservation and Centralized Management System

Don Y.J Holdings

- SOFTWARE TESTING DESIGN -

Test Scope:

The major scope of the current testing design is to make sure that the online reservation and centralized management system for Don Y.J Holdings meets the proper requirements stated by the company. Once the entire web application is developed, it would be ready for a complete software testing procedure to ensure that the functionalities are properly functional.

Testing of any software application would include validation and verification of multiple system factors. This also includes the different types of functionalities and features implemented during the software development phase. In case the software system would still contain certain errors missed during implementation, the testing process would assist to identify them.

Test Objectives:

- Verify if initial system analysis and development methodologies are followed.
- Verify whether the initial system requirements correspond with the project progress.
- Verify if the system design solves the initial problems stated by the company.
- Check the efficiency and effectiveness of the entire project phases.

Test Approach:

The current software testing design for the online reservation and centralized management system for Don Y.J Holdings should be done in a step-by-step method in order to achieve accurate results. It is essential that the software is tested in application user's perspective, so that the system testing would be carried out just as being deployed in the real world.

Software testing is an essential software development phase to ensure that the application being developed actually corresponds with the initial requirements. Moreover, software testing confirms that the functionalities included in the application are of proper standards and works as intended. The most important purpose of software testing is to identify if the application functions do not consist of critical errors. In order to identify system errors. The application's key and extra functionalities can be professionally tested. It is also essential to document the testing procedures and approaches in order to evaluate them later. The documentation of the application's testing phase is known as the "software testing design" document.

In case the software testing procedure identifies any critical or minor errors in the application, it would always be a professional practice to fix the errors and optimize the final system before it is delivered to the client or deployed in the environment. Currently, there are various manual and automated methodologies of software testing. Properly tested software is a universal method to identify that it would include proper efficiency and better user experience. In case any software is skipped through the testing phase and is deployed in the user environment, it could cause major catastrophes to multiple factors including the client businesses or customer experiences.

Considering software testing methodologies, they could be mainly divided into two types namely manual testing and automated testing. Manual testing has been the traditional methodology of software testing which requires a longer time, but every small aspect of the software would be tested until the developers are satisfied. Currently, automated testing methodologies are attracting various software projects, sue to their nature of easier handling. Automated testing could be time saving and are easier to handle. Automated testing methodologies could include special frameworks specifically designed to the purposed of software testing designs.



Figure 88: Software Testing

Functional Testing:

Functional testing is the testing procedure of verifying the software application against the initial functional requirements of any software development project. Functional testing required the application's every single function to be tested for proper functionality and effectiveness. The universal methodology of carrying out functional testing is through three processes of input, verification and output. The results of each test would then be evaluated against the initially collected requirements regarding the functionalities of the application. (W3Softech, 2019)



Figure 89: Functional Testing

The diagram below illustrates the procedure of a typical functional testing;

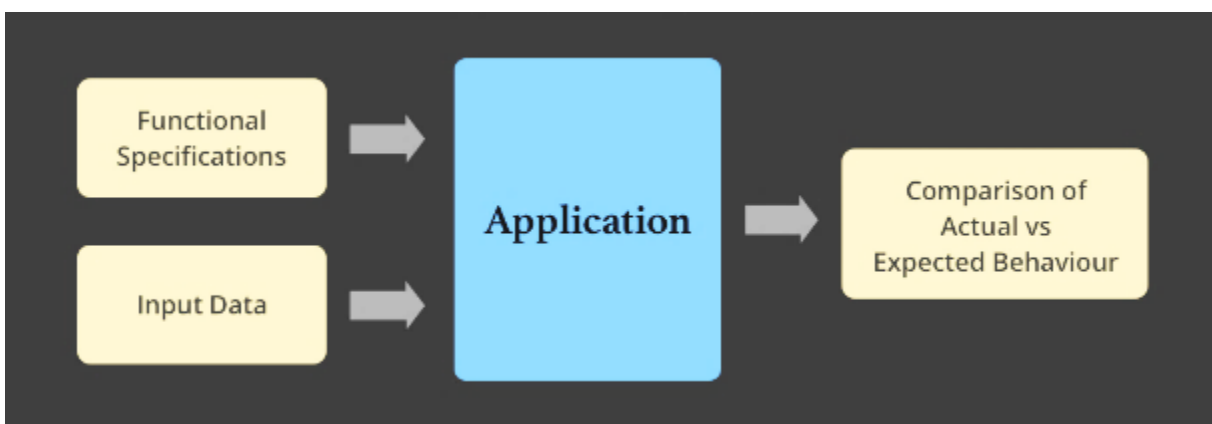


Figure 90: Functional testing procedure illustration

Unit Testing:

Unit testing is a function software testing methodology where the application functionalities are divided into smaller aspects called “units” in order to be tested. Each unit would be tested individually for its specific functionality. The major intention of unit testing is to ensure if the specific unit function is properly functional, as intended. (Hamilton, 2021)

Integration Testing:

Integration testing is considered as the second testing phase following unit testing. Multiple units of the application would be tested together in this phase. This would reveal any errors when the units are working together. Any issues during the functional integration of the units could be tested through this phase. (JavaTpoint, 2021)

Black Box Testing:

Black box testing is an integration testing type where the application functionalities are tested without the code-based knowledge of the system. The tester doesn't require to have the knowledge of the application's internal coding structures, back-end links or scripts. The testing majorly focused on providing an input and evaluating the potential outcomes. (Hamilton, 2021)

White Box Testing:

White box testing is another integration testing type where the application's coding structure is tested for proper flows of functionalities. The functionalities are based with proper inputs and outputs, where the output is analysed along with the function's coding structure. This testing type ensures that the actual functionality corresponds with the code structures. (Hamilton, 2021)

Integration Testing Template:

Table 11: Integration testing template

Test Description				
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome Result (Pass/Fail)

Integration Testing Targets:

Table 12: Integration testing targets

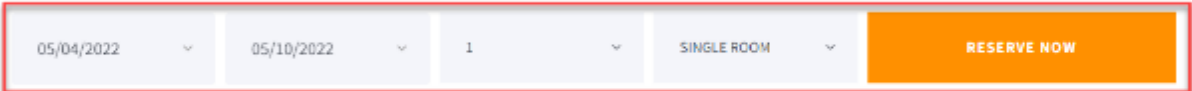
Integration Testing Targets		
Online Reservation and Centralized Management System		
Test Case ID	Test Target	Test Date
01	Room reservation.	21/04/2022
02	Guest user registration.	10/02/2022
03	Guest user login.	10/02/2022
04	Reservation system administrator login.	22/04/2022
05	Add and manage room types.	23/04/2022
06	Add and manage rooms.	24/04/2022

07	View and manage reservations.	22/04/2022
08	Generate and print booking report.	25/04/2022
09	Management system login.	23/04/2022
10	Add and manage departments.	21/04/2022
11	Add and manage employees.	25/04/2022
12	Add and manage employee payroll.	26/04/2022
13	Generate employee salary invoice.	27/04/2022
14	Create employee leave requests.	28/04/2022
15	View and manage employee leave requests.	28/04/2022

Table 13: Integration test case 1

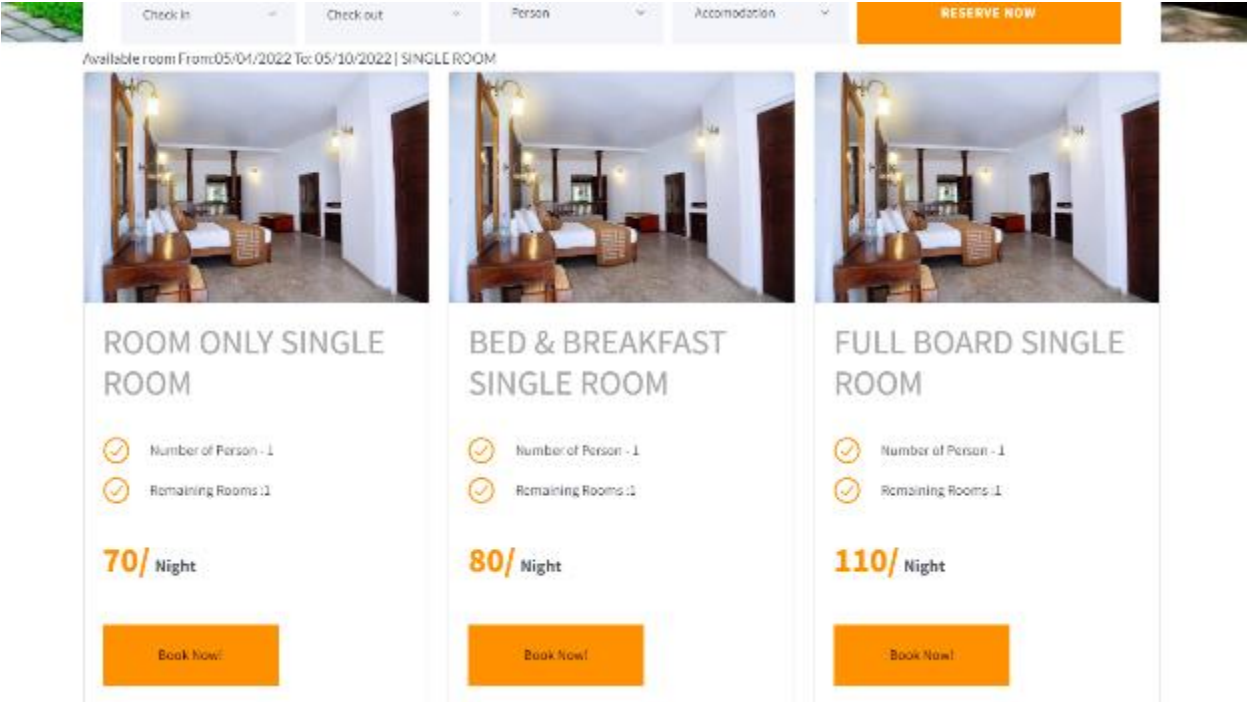
Test Target		Room reservation.		
Test Description		Search room availability.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
01	Check in: 05/04/2022 Checkout: 05/10/2022 Guests: 1 Room Type: Single	View available room details based on the input data.	Viewed available room details based on the input data.	PASS

Evidence:



The screenshot shows a reservation form with the following fields: Check in (05/04/2022), Check out (05/10/2022), Person (1), and Accommodation (SINGLE ROOM). A red box highlights these fields and the 'RESERVE NOW' button.

Figure 91: Integration test case 1 - Evidence #1



The screenshot shows the results of the room search. It displays three room options, each with a photo, title, details, price, and a 'Book Now!' button. The details include the number of persons and remaining rooms.

Room Type	Price per Night
ROOM ONLY SINGLE ROOM	70/ Night
BED & BREAKFAST SINGLE ROOM	80/ Night
FULL BOARD SINGLE ROOM	110/ Night

Figure 92: Integration test case 1 - Evidence #2

Table 14: Integration test case 1.1

Test Target		Room reservation.		
Test Description		Confirm reservation.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
1.1	User login with account or creates new account.	Confirm reservation with billing details and “Booking is successfully submitted” message	Confirm reservation with billing details and “Booking is successfully submitted” message	PASS

Evidence:

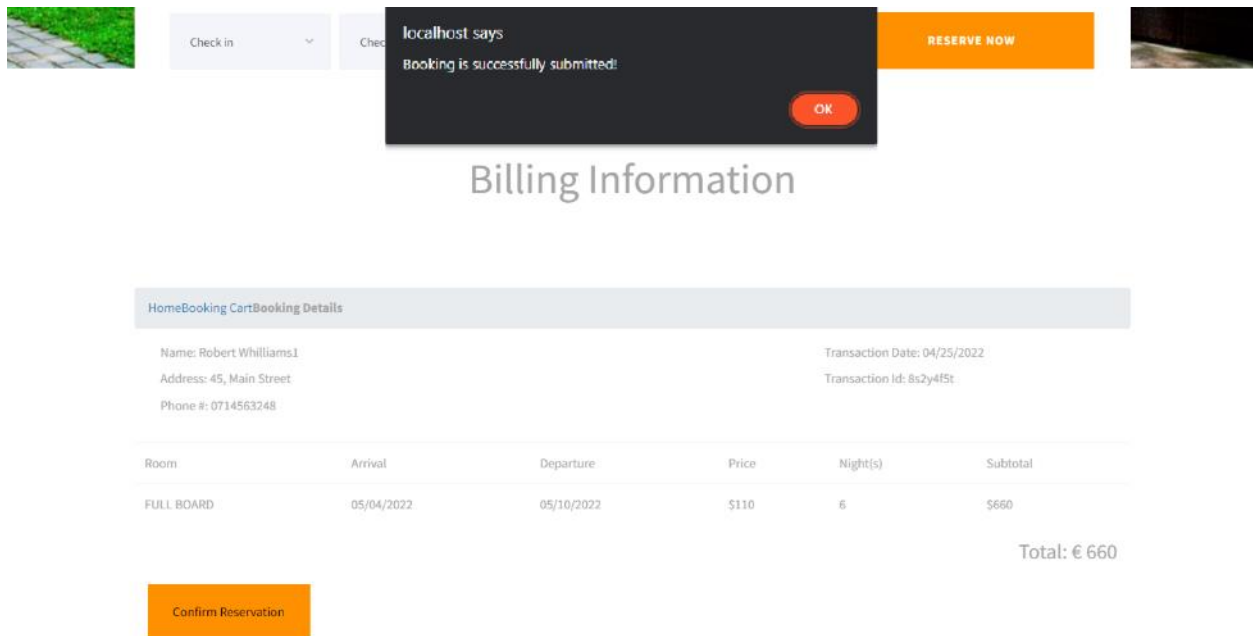


Figure 93: Integration test case 1.1 - Evidence

Table 15: Integration test case 2

Test Description		Guest user registration.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
02	First Name: John Last Name: Howard City: Colombo Address: 56/87, Greens Avenue, Main Street, Colombo 07 Date of Birth: 05/02/2001 Mobile: 0789510237 Nationality: Sri Lankan Email: johnhoward@gmail.com Username: johnhoward123 Password: johnhowardlenn123 Postal Code: 00700	Guest record successfully created in the user database table.	Guest record successfully created in the user database table.	PASS

Evidence:

Figure 94: Integration test case 2 - Evidence #1

Eventually, the entered guest record would be saved in the user database, linked with a unique ID for specific user type, as shown below.

GUESTID	REFNO	G_FNAME	G_LNAME	G_CITY	G_ADDRESS	DBIRTH	G_PHONE	G_NATIONALITY	G_EMAIL	G_UNAME	G_PASS	ZIP
11122	0	Robert	Williams		45, Main Street	2002-04-11	0714563248	Sri Lankan	testuser@gmail.com	test1	7288edd0fc3ffcbe93a0c06e3568e28521687bc	20100
11125	0	John	Howard	Colombo	56/87, Greens Avenue, Main Street, Colombo 07	2001-05-02	0789510237	Sri Lankan	johnhoward@gmail.com	johnhoward123	c52d5cd7499db072fde76702553fb1caa80d03d6	700

Figure 95: Integration test case 2 - Evidence #2

Table 16: Integration test case 3

Test Description		Guest user login.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
03	Username: johnhoward123 Password: johnhowardlenn123	“Login Successful” message	“Login Successful” message	PASS

Evidence:

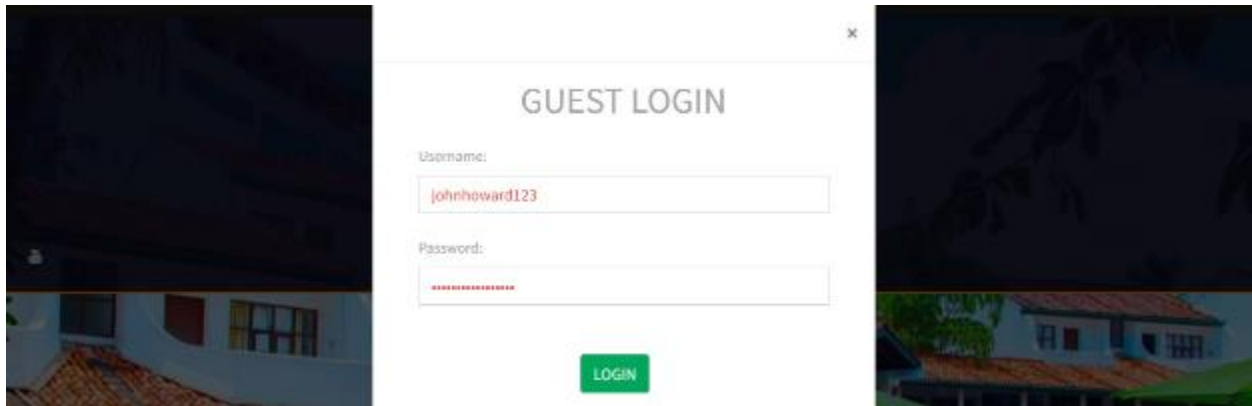


Figure 96: Integration test case 3 - Evidence #1

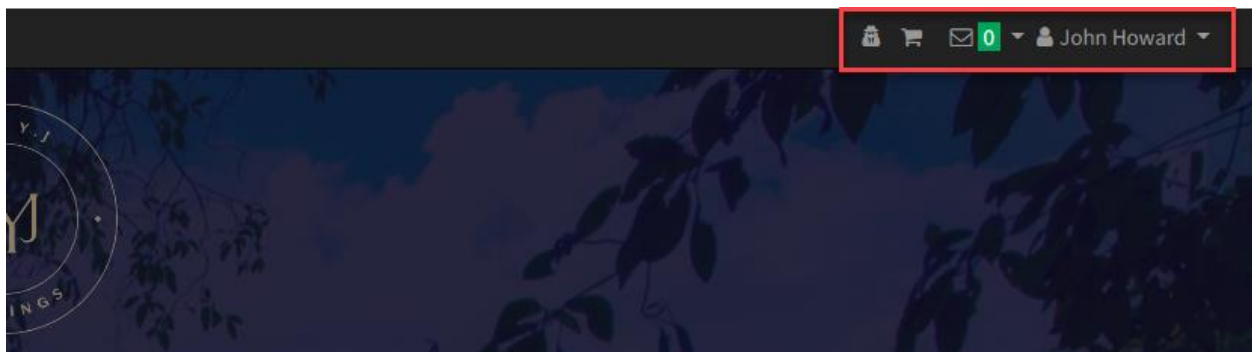


Figure 97: Integration test case 3 - Evidence #2

Table 17: Integration test case 4

Test Description		Reservation system administrator login.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
04	Username: admin Password: admin	Successful login and redirection to dashboard.	Successful login and redirection to dashboard.	PASS

Evidence:

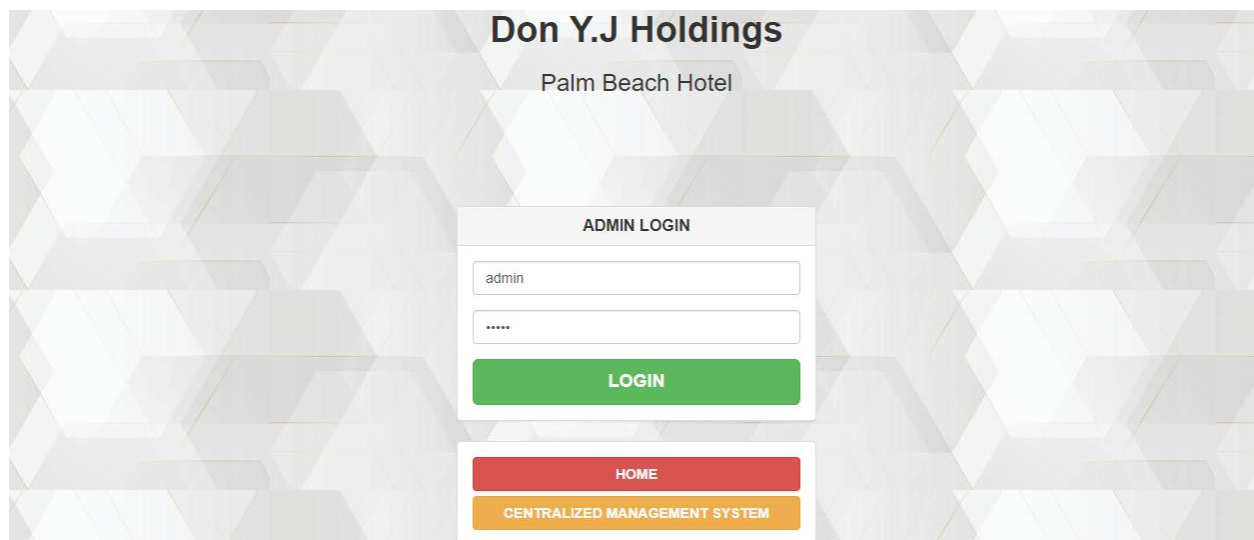


Figure 98: Integration test case 4 - Evidence #1

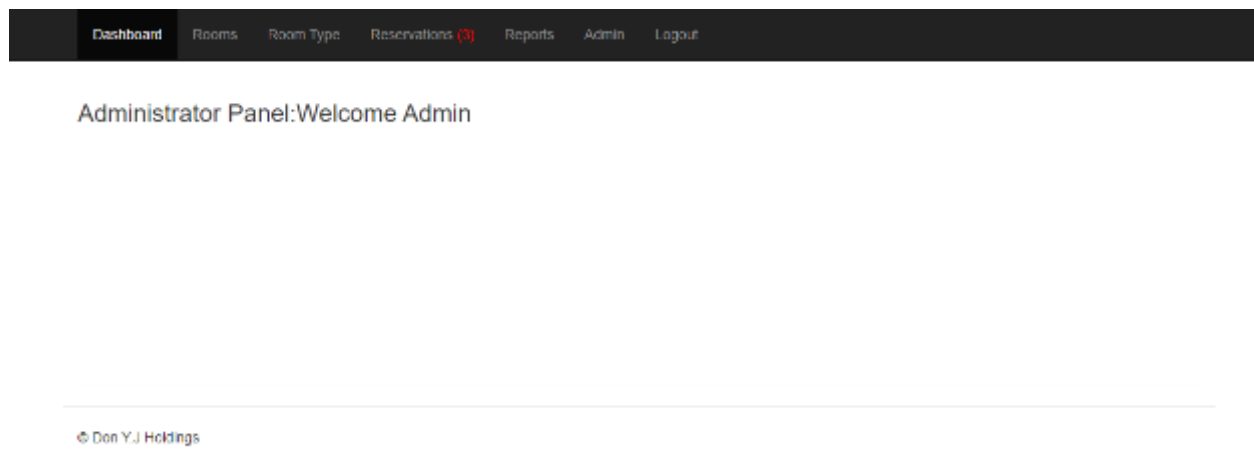


Figure 99: Integration test case 4 - Evidence #2

Table 18: Integration test case 5

Test Target		Add and manage room types.		
Test Description		Create new room type.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
05	Name: TRIPLE ROOM	Room type created as database record.	Room type created as database record.	PASS

Evidence:

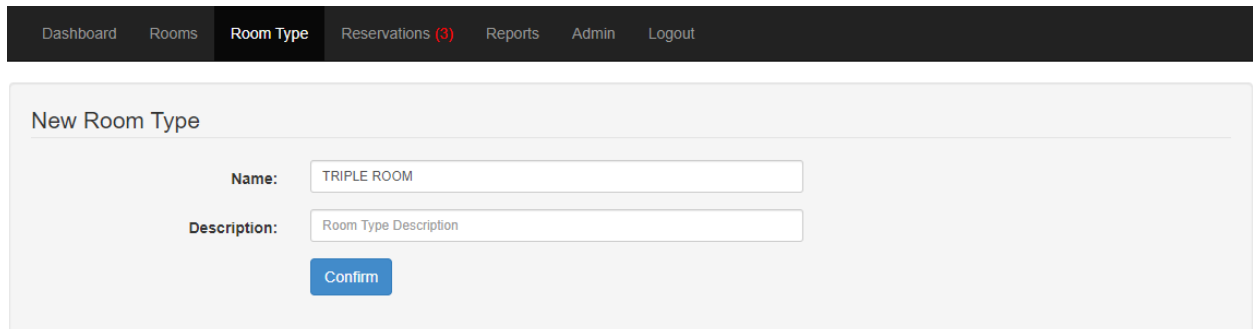
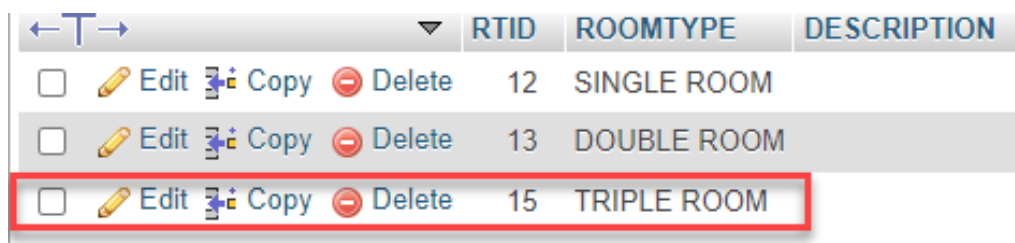


Figure 100: Integration test case 5 - Evidence #1



	RTID	ROOMTYPE	DESCRIPTION
<input type="checkbox"/> Edit Copy Delete	12	SINGLE ROOM	
<input type="checkbox"/> Edit Copy Delete	13	DOUBLE ROOM	
<input type="checkbox"/> Edit Copy Delete	15	TRIPLE ROOM	

Figure 101: Integration test case 5 - Evidence #2

Table 19: Integration test case 5.1

Test Target		Add and manage room types.		
Test Description		View available room types.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
5.1	-	View existing room type records.	View existing room type records.	PASS

Evidence:

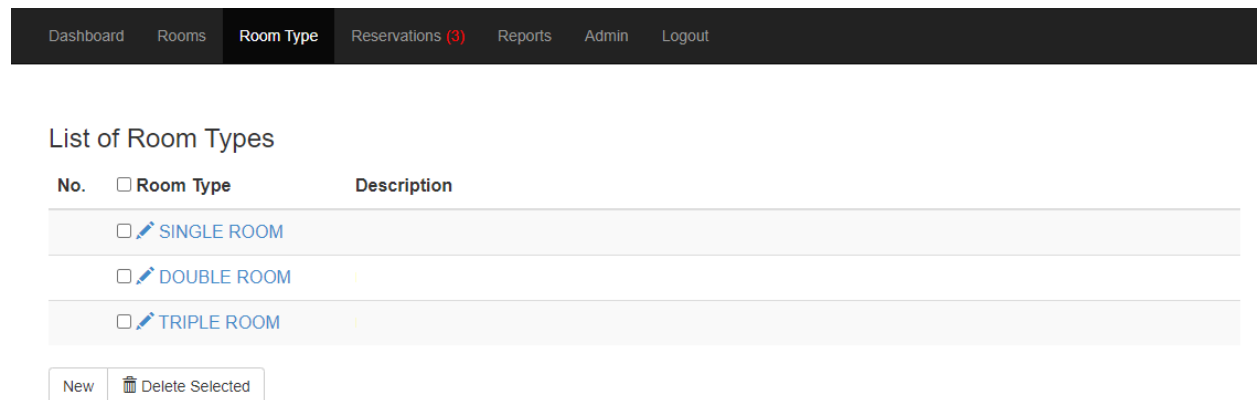


Figure 102: Integration test case 5.1 - Evidence #1

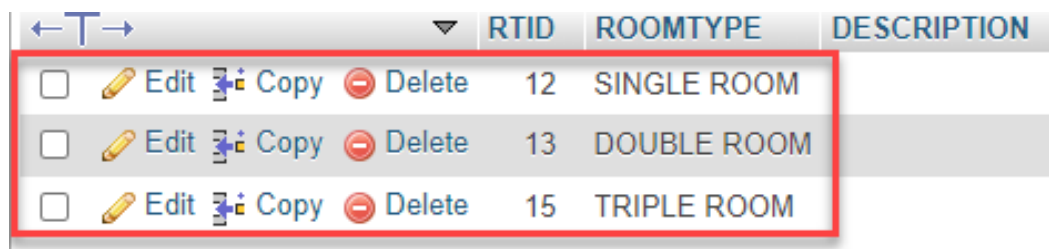


Figure 103: Integration test case 5.1 - Evidence #2

Table 20: Integration test case 6

Test Target		Add and manage rooms.		
Test Description		Add new room.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
06	Name: SUPERIOR TWIN ROOM ONLY Room Type: DOUBLE ROOM Number of Guests: 2 Price: \$85 Image: Twin Room	Room record added as database record.	Room record added as database record.	PASS

Evidence:

The screenshot shows a web application interface for adding a new room. The navigation bar includes links for Dashboard, Rooms (active), Room Type, Reservations (2), Reports, Admin, and Logout. The 'New Room' form contains the following fields and values:

- Name:** SUPERIOR TWIN ROOM ONLY
- Room Type:** DOUBLE ROOM
- Description:** Description
- Number of Person:** 2
- Price (\$):** 85
- Upload Image:** Choose File, superior-twin.jpg

A blue 'Save' button is located at the bottom of the form.

Figure 104: Integration test case 6 - Evidence #1

		ROOMID	ROOMNUM	ACCOMID	ROOM	ROOMDESC	NUMPERSON	PRICE	ROOMIMAGE	OROOMNUM
<input type="checkbox"/>	Edit Copy Delete	17	1	12	ROOM ONLY		1	70	rooms/202204240643_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	18	1	12	BED & BREAKFAST		1	80	rooms/202204240644_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	19	0	12	HALF BOARD		1	96	rooms/202204240645_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	20	1	12	FULL BOARD		1	110	rooms/202204240646_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	29	1	13	SUPERIOR TWIN ROOM ONLY		2	85	rooms/202204261005_superior-twin.jpg	1

Figure 105: Integration test case 6 - Evidence #2

Table 21: Integration test case 6.1

Test Target		Add and manage rooms.		
Test Description		View available rooms.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
6.1	-	View existing room records.	View existing room records.	PASS

Evidence:

Dashboard	Rooms	Room Type	Reservations (2)	Reports	Admin	Logout
-----------	--------------	-----------	------------------	---------	-------	--------

List of Rooms






No.	<input type="checkbox"/> Image	Room	Room Type	Person	Price
<input type="checkbox"/>		ROOM ONLY ()	SINGLE ROOM	1	\$70
<input type="checkbox"/>		BED & BREAKFAST ()	SINGLE ROOM	1	\$80
<input type="checkbox"/>		HALF BOARD ()	SINGLE ROOM	1	\$96
<input type="checkbox"/>		FULL BOARD ()	SINGLE ROOM	1	\$110
<input type="checkbox"/>		SUPERIOR TWIN ROOM ONLY ()	DOUBLE ROOM	2	\$85

Figure 106: Integration test case 6.1 - Evidence #1

		ROOMID	ROOMNUM	ACCOMID	ROOM	ROOMDESC	NUMPERSON	PRICE	ROOMIMAGE	OROOMNUM
<input type="checkbox"/>	Edit Copy Delete	17	1	12	ROOM ONLY		1	70	rooms/202204240643_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	18	1	12	BED & BREAKFAST		1	80	rooms/202204240644_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	19	0	12	HALF BOARD		1	96	rooms/202204240645_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	20	1	12	FULL BOARD		1	110	rooms/202204240646_Single.jpg	1
<input type="checkbox"/>	Edit Copy Delete	29	1	13	SUPERIOR TWIN ROOM ONLY		2	85	rooms/202204261005_superior-twin.jpg	1

Figure 107: Integration test case 6.1 - Evidence #2

Table 22: Integration test case 6.2

Test Target		Add and manage rooms.		
Test Description		Edit existing room record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
6.2	Name: SUPERIOR TWIN ROOM OFFER Room Type: DOUBLE ROOM Description: Limited Time Offer Price: \$75	Update selected room record changes.	Update selected room record changes.	PASS

Evidence:

The screenshot shows the 'Edit Room' form with the following fields highlighted by red boxes:

- Name:** SUPERIOR TWIN ROOM OFFER
- Room Type:** DOUBLE ROOM (selected from a dropdown)
- Description:** Limited Time Offer
- Number of Person:** 2
- Price (\$):** 75

The 'Upload Image' section shows a 'Choose File' button and the text 'No file chosen'.

Figure 108: Integration test case 6.2 - Evidence #1

The screenshot shows the 'List of Rooms' table with the following data:

No.	Image	Room	Room Type	Person	Price
		ROOM ONLY ()	SINGLE ROOM	1	\$70
		BED & BREAKFAST ()	SINGLE ROOM	1	\$80
		HALF BOARD ()	SINGLE ROOM	1	\$96
		FULL BOARD ()	SINGLE ROOM	1	\$110
		SUPERIOR TWIN ROOM OFFER (Limited Time Offer)	DOUBLE ROOM	2	\$75

The last row, representing the 'SUPERIOR TWIN ROOM OFFER', is highlighted with a red box.

Figure 109: Integration test case 6.2 - Evidence #2

Table 23: Integration test case 6.3

Test Target		Add and manage rooms.		
Test Description		Delete existing room record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
6.3	-	Delete selected room record from database.	Delete selected room record from database.	PASS

Evidence:

The screenshot shows a web application interface for managing rooms. At the top, there is a navigation bar with links: Dashboard, Rooms (active), Room Type, Reservations (2), Reports, Admin, and Logout. Below the navigation bar, the title 'List of Rooms' is displayed. A table lists the following rooms:

No.	Image	Room	Room Type	Person	Price
<input type="checkbox"/>		ROOM ONLY ()	SINGLE ROOM	1	\$70
<input type="checkbox"/>		BED & BREAKFAST ()	SINGLE ROOM	1	\$80
<input type="checkbox"/>		HALF BOARD ()	SINGLE ROOM	1	\$96
<input type="checkbox"/>		FULL BOARD ()	SINGLE ROOM	1	\$110
<input checked="" type="checkbox"/>		SUPERIOR TWIN ROOM OFFER (Limited Time Offer)	DOUBLE ROOM	2	\$75

Below the table, there are two buttons: 'New Room' and 'Delete Selected' (highlighted with a yellow box). A red box at the top right of the table area contains the message 'Room Deleted!'.

Figure 110: Integration test case 6.3 - Evidence #1

			ROOMID	ROOMNUM	ACCOMID	ROOM	ROOMDESC	NUMPERSON	PRICE	ROOMIMAGE	ROOMNUM
<input type="checkbox"/>				17	1	12	ROOM ONLY	1	70	rooms/202204240643_Single.jpg	1
<input type="checkbox"/>				18	1	12	BED & BREAKFAST	1	80	rooms/202204240644_Single.jpg	1
<input type="checkbox"/>				19	0	12	HALF BOARD	1	96	rooms/202204240645_Single.jpg	1
<input type="checkbox"/>				20	1	12	FULL BOARD	1	110	rooms/202204240646_Single.jpg	1

Below the table, there is a toolbar with the following options: ☐ Check all, With selected: Edit, Copy, Delete, Export.

Figure 111: Integration test case 6.3 - Evidence #2

Table 24: Integration test case 7

Test Target		View and manage reservations.		
Test Description		View existing room reservation list and details.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
07	-	View list of existing reservations and selected details.	View list of existing reservations and selected details.	PASS

Evidence:

Dashboard	Rooms	Room Type	Reservations (2)	Reports	Admin	Logout
-----------	-------	-----------	------------------	---------	-------	--------

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Whilliams	2022-04-25 02:43:02	bjnchkbx	1	€660	Pending	View Cancel Confirm Delete Edit
	Robert Whilliams	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	View Cancel Confirm Delete Edit
	John Howard	2022-04-25 03:01:27	sefg53t7	1	€540	Cancelled	View Cancel Confirm Delete Edit
	Robert Whilliams	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	View Delete Edit

Figure 112: Integration test case 7 - Evidence #1

Dashboard	Rooms	Room Type	Reservations (2)	Reports	Admin	Logout
-----------	-------	-----------	------------------	---------	-------	--------

Guest Information

FIRSTNAME: Robert

LASTNAME: Whilliams

ADDRESS: 45, Main Street

Reservation



FULL BOARD [SINGLE ROOM]

ARRIVAL: 05/04/2022

DEPARTURE: 05/10/2022

Night(s): 6

PRICE: \$660

[View Rooms](#)

Figure 113: Integration test case 7 - Evidence #2

Table 25: Integration test case 7.1

Test Target		View and manage reservations.		
Test Description		Confirm reservation and update status.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
7.1	-	Confirm selected reservation.	Confirm selected reservation.	PASS

Evidence:

The screenshot shows a web application interface for managing reservations. At the top is a navigation bar with links: Dashboard, Rooms, Room Type, Reservations (2), Reports, Admin, and Logout. Below the navigation bar, the page is titled 'Guest Information' and 'Reservation View Rooms'. On the left, there are input fields for 'FIRSTNAME' (Robert), 'LASTNAME' (Williams), and 'ADDRESS' (45, Main Street). In the center, there is a room image and reservation details: 'FULL BOARD [SINGLE ROOM]', 'ARRIVAL: 05/04/2022', 'DEPARTURE: 05/10/2022', 'Night(s): 6', and 'PRICE: \$660'. At the bottom left is a '+ Back' button, and at the bottom right is a 'Confirm' button highlighted with a red box.

Figure 114: Integration test case 7.1 - Evidence #1

The screenshot shows the same web application interface after a reservation update. A green message bar at the top says 'Reservation Updated successfully!'. Below it is a table listing reservations. The table has columns: #, Guest, Transaction Date, Confirmation Code, Total Rooms, Total Price, Status, and Action. The third row, for Robert Williams, is highlighted with a red box, showing the reservation is now 'Confirmed' with a price of €660. The 'Action' column for this row includes buttons for View, Cancel, Confirm, Delete, and Edit.

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Williams	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	View Cancel Confirm Delete Edit
	John Howard	2022-04-25 03:01:27	sefg53t7	1	€540	Cancelled	View Cancel Confirm Delete Edit
	Robert Williams	2022-04-25 02:43:02	bjnchkbx	1	€660	Confirmed	View Cancel Check in Delete Edit
	Robert Williams	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	View Delete Edit

Figure 115: Integration test case 7.1 - Evidence #2

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Whilliams	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	View Cancel Confirm Delete Edit
	John Howard	2022-04-25 03:01:27	sefg53t7	1	€540	Cancelled	View Cancel Confirm Delete Edit
	Robert Whilliams	2022-04-25 02:43:02	bjnchkbx	1	€660	Checkedin	View Check out Delete Edit
	Robert Whilliams	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	View Delete Edit

Figure 116: Integration test case 7.1 - Evidence #3

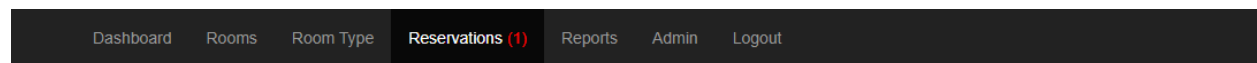
#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Whilliams	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	View Cancel Confirm Delete Edit
	John Howard	2022-04-25 03:01:27	sefg53t7	1	€540	Cancelled	View Cancel Confirm Delete Edit
	Robert Whilliams	2022-04-25 02:43:02	bjnchkbx	1	€660	Checkedout	View Delete Edit
	Robert Whilliams	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	View Delete Edit

Figure 117: Integration test case 7.1 - Evidence #4

Table 26: Integration test case 7.2

Test Target		View and manage reservations.		
Test Description		Edit reservation guest information.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
7.2	Last Name: Williamson Address: 68, Central Road	Update selected reservation guest information.	Update selected reservation guest information.	PASS

Evidence:



Edit Reservation

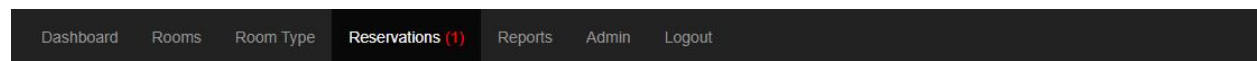
Guest Information

FIRST NAME:

LAST NAME:

ADDRESS:

Figure 118: Integration test case 7.2 - Evidence #1



Guest Information

FIRSTNAME : Robert

LASTNAME : Williamson

ADDRESS : 68, Central Street



Reservation View Rooms

FULL BOARD [SINGLE ROOM]

ARRIVAL: 05/04/2022

DEPARTURE: 05/10/2022

Night(s): 6

PRICE: \$660

Figure 119: Integration test case 7.2 - Evidence #2

Table 27: Integration test case 7.3

Test Target		View and manage reservations.		
Test Description		Delete existing reservation record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
7.3	-	Delete selected reservation record.	Delete selected reservation record.	PASS

Evidence:

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Whilliamson	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	View Cancel Confirm Delete Edit
	John Howard	2022-04-25 03:01:27	sefg53t7	1	€540	Cancelled	View Cancel Confirm Delete Edit
	Robert Whilliamson	2022-04-25 02:43:02	bjnchkbx	1	€660	Checkedout	View Delete Edit
	Robert Whilliamson	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	View Delete Edit

Figure 120: Integration test case 7.3 - Evidence #1

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Whilliamson	2022-04-24 12:53:41	uzhm7zpm	3	€528	Pending	View Cancel Confirm Delete Edit
	Robert Whilliamson	2022-04-25 02:43:02	bjnchkbx	1	€660	Checkedout	View Delete Edit
	Robert Whilliamson	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	View Delete Edit

Figure 121: Integration test case 7.3 - Evidence #2

Table 28: Integration test case 8

Test Target		Generate and print booking report.		
Test Description		Generate existing booking report using reservation filters.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Outcome (Pass/Fail)
08	Status: Checked-out Check-in: 2022-04-24 Check-out: 2022-04-25	Generate room booking reports within selection.	Generate room booking reports within selection.	PASS

Evidence:

Dashboard
Rooms
Room Type
Reservations (0)
Reports
Admin
Logout

Room Booking Report
Date: 04/26/2022

Status
Checkedin
Checkedout

Checkedout
Checkedin Date :2022-04-24 Checkedout Date :2022-04-26

Guest	Room	Price	Arrival	Departure	Night(s)	Subtotal
Robert Whilliamson	SINGLE ROOM [HALF BOARD]	\$ 96	04/24/2022	04/24/2022	1	\$ 96
Robert Whilliamson	DOUBLE ROOM [FULL BOARD]	\$ 150	04/24/2022	04/24/2022	1	\$ 150
Robert Whilliamson	TRIPLE ROOM [HALF BOARD]	\$ 168	04/24/2022	04/24/2022	1	\$ 168
Robert Whilliamson	TRIPLE ROOM [FULL BOARD]	\$ 210	04/24/2022	04/24/2022	1	\$ 210

Total Amount

Subtotal: \$ 624

Figure 122: Integration test case 8 - Evidence #1

Table 29: Integration test case 9

Test Target		Management system login.		
Test Description		Administrator login.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
09	Username: admin Password: admin	Successful login and redirection to admin dashboard.	Successful login and redirection to admin dashboard.	PASS

Evidence:

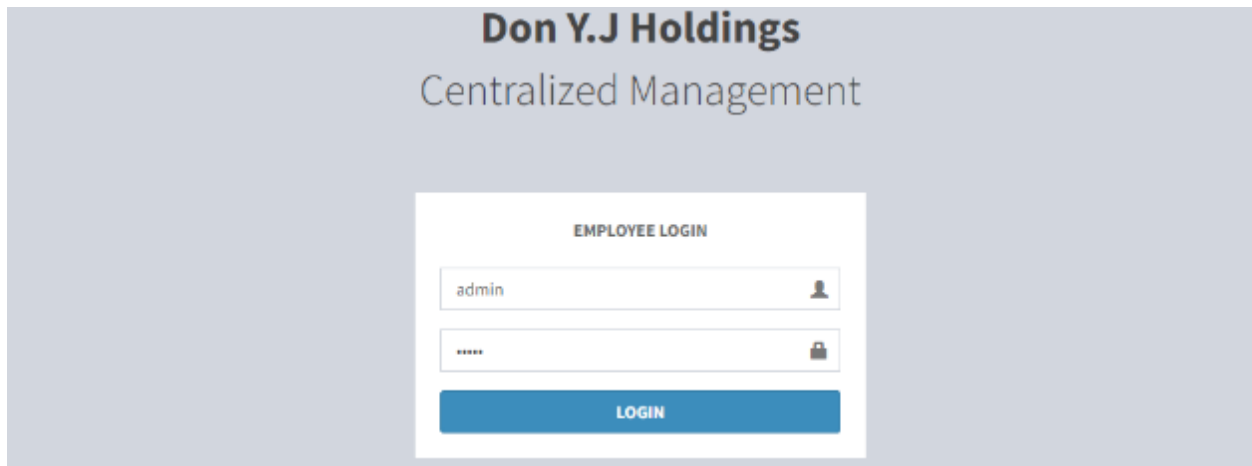


Figure 123: Integration test case 9 - Evidence #1

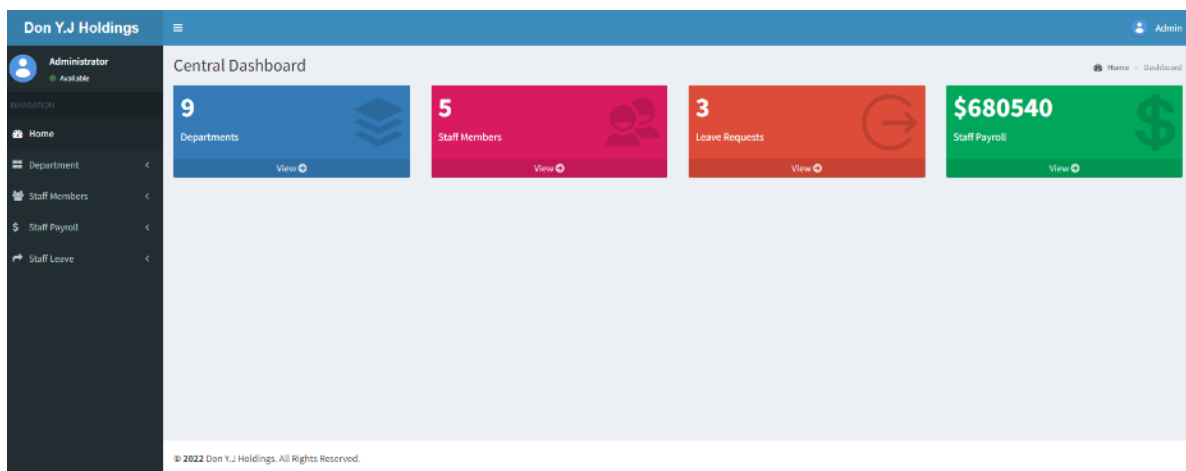


Figure 124: Integration test case 9 - Evidence #2

Table 30: Integration test case 9.1

Test Target		Management system login.		
Test Description		Employee login.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
9.1	Username: roland@yahoo.com Password: roland123	Successful login and redirection to employee dashboard.	Successful login and redirection to employee dashboard.	PASS

Evidence:

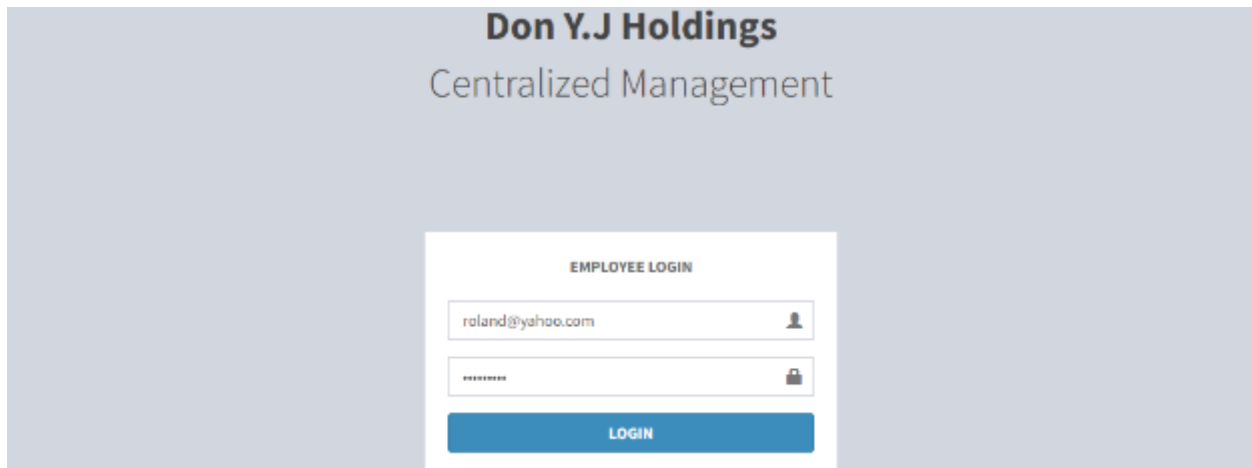


Figure 125: Integration test case 9.1 - Evidence #1

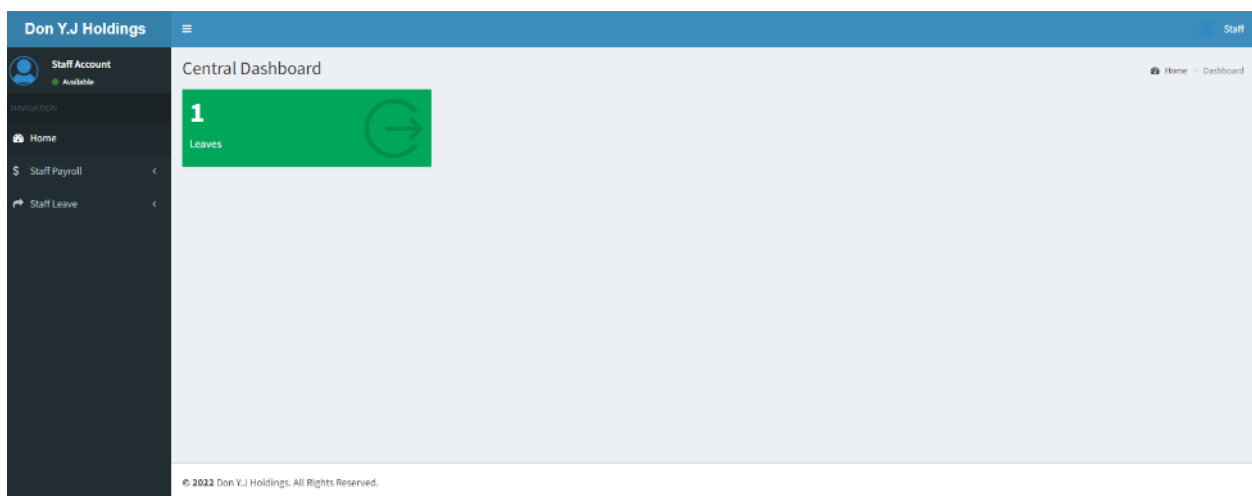


Figure 126: Integration test case 9.1 - Evidence #2

Table 31: Integration test case 10

Test Target		Add and manage departments.		
Test Description		Add new department.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
10	Name: Engineering	Department added as database record.	Department added as database record.	PASS

Evidence:

Don Y.J Holdings

Administrator Available

NAVIGATION

- Home
- Department
- Staff Members
- Staff Payroll
- Staff Leave

Departments

Home > Departments > Add Department

Add Department

Department Name

Engineering

Submit

© 2022 Don Y.J Holdings. All Rights Reserved.

Figure 127: Integration test case 10 - Evidence #1

					id	department_name
<input type="checkbox"/>		Edit		Copy		Delete
					7	Administration
<input type="checkbox"/>		Edit		Copy		Delete
					8	Human Resources (HR)
<input type="checkbox"/>		Edit		Copy		Delete
					9	Security
<input type="checkbox"/>		Edit		Copy		Delete
					16	Engineering

Figure 128: Integration test case 10 - Evidence #2

Table 32: Integration test case 10.1

Test Target		Add and manage departments.		
Test Description		View list of existing departments.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
10.1	-	View list of existing departments.	View list of existing departments.	PASS

Evidence:

#	Department Name	Actions
1	Administration	Edit Delete
2	Human Resources (HR)	Edit Delete
3	Security	Edit Delete
4	Information Technology (IT)	Edit Delete
5	Sales and Marketing	Edit Delete
6	Accounts and Finance	Edit Delete
7	Food and Beverage	Edit Delete
8	Housekeeping	Edit Delete
9	Front Office	Edit Delete
10	Engineering	Edit Delete

Figure 129: Integration test case 10.1 - Evidence #1

id	department_name
7	Administration
8	Human Resources (HR)
9	Security
10	Information Technology (IT)
11	Sales and Marketing
12	Accounts and Finance
13	Food and Beverage
14	Housekeeping
15	Front Office
16	Engineering

Figure 130: Integration test case 10.1 - Evidence #2

Table 33: Integration test case 10.2

Test Target		Add and manage departments.		
Test Description		Update existing department record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
10.2	Name: Engineering and Maintenance	Update selected department record.	Update selected department record.	PASS

Evidence:

The screenshot shows the 'Don Y.J Holdings' web application interface. On the left is a dark sidebar with a navigation menu containing 'Home', 'Department', 'Staff Members', 'Staff Payroll', and 'Staff Leave'. The main content area is titled 'Departments' and shows an 'Edit Department' form. The form has a 'Department Name' input field with the text 'Engineering and Maintenance' and a green 'Update' button. The top header bar is blue and contains the company name 'Don Y.J Holdings' and a user profile 'Admin'. The footer of the page states '© 2022 Don Y.J Holdings. All Rights Reserved.'

Figure 131: Integration test case 10.2 - Evidence #1

	id	department_name
<input type="checkbox"/> Edit Copy Delete	7	Administration
<input type="checkbox"/> Edit Copy Delete	8	Human Resources (HR)
<input type="checkbox"/> Edit Copy Delete	9	Security
<input type="checkbox"/> Edit Copy Delete	16	Engineering and Maintenance

Figure 132: Integration test case 10.2 - Evidence #2

Table 34: Integration test case 10.3

Test Target		Add and manage departments.		
Test Description		Delete existing department record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
10.3	-	Delete selected department record.	Delete selected department record.	PASS

Evidence:

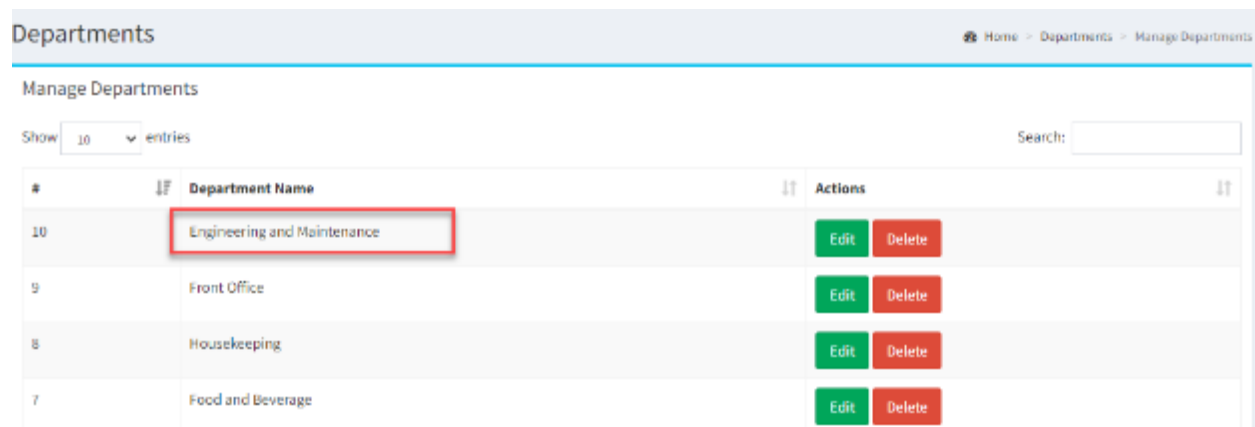


Figure 133: Integration test case 10.3 - Evidence #1

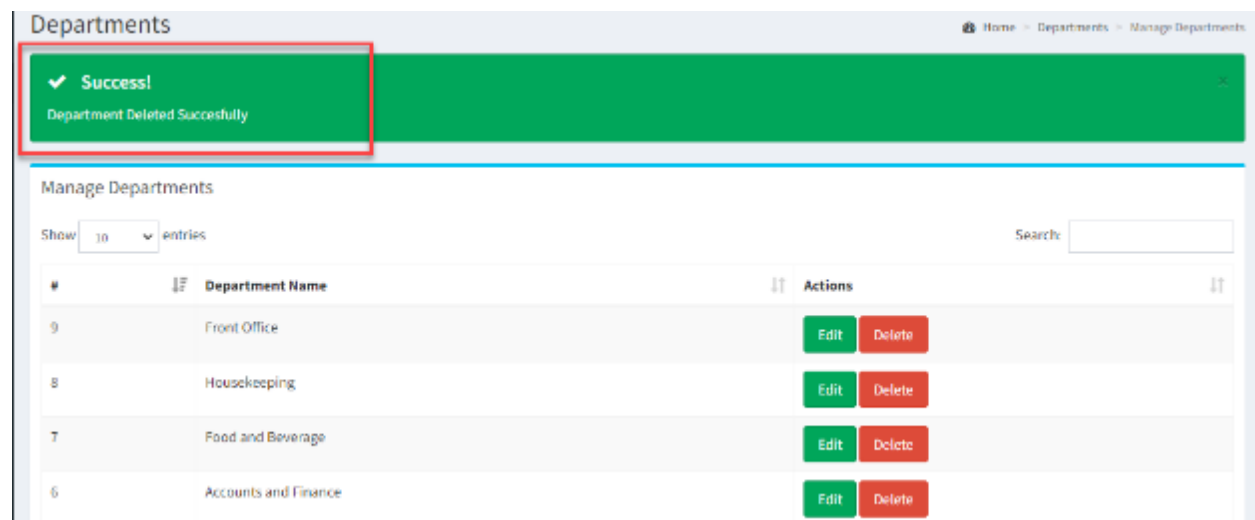


Figure 134: Integration test case 10.3 - Evidence #2

Table 35: Integration test case 11

Test Target		Add and manage employees.		
Test Description		Add new employee.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
11	Name: Taylor James Department: Front Office Gender: Male Email: jamestaylor34@yahoo.com Mobile: 9412395167 Date of Birth: 08/11/1999 Reg Date: 08/02/2022 City: Dehiwala State: Western Country: Sri Lanka Password: jamestayloe34 Address: No. 56, Central Avenue, Dehiwala	Employee added as database record.	Employee added as database record.	PASS

Evidence:

Staff Management Home > Staff Management > Add Staff

Add Staff

Full Name Taylor James	Department Front Office
Gender Select	Email jamestaylor34@yahoo.com
Mobile 941237951	Photo Choose File avatar7.1ee966d056f0.png
Date of Birth 08/11/1999	Date of Joining 08/02/2022
City Dehiwala	State Western
Country Sri Lanka	Password *****
Address No. 56, Central Avenue, Dehiwala	

Submit

Figure 135: Integration test case 11 - Evidence #1

	id	staff_name	gender	email	mobile	dob	doj	address	city	state	country	department_id
	11	Robert Wileys	Male	robertwileys@gmail.com	7891230458	1988-11-09	2015-10-08	56/98, First Street	Colombo 02	Western	Sri Lanka	7
	12	Anna Statham	Female	annastata@yahoo.com	4596320178	1992-11-12	2020-08-05	789, Second Cross Street	Galle	Southern	Sri Lanka	8
	13	David Jones	Male	david@gmail.com	1254630879	1999-07-14	2019-02-16	56/67, Cross Street	Avissawella	Western	Sri Lanka	10
	14	Jennifer Larry	Female	jennifer@outlook.com	2354187924	1994-06-04	2020-07-15	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	12
	15	Roland Wills	Male	roland@yahoo.com	7981354207	1998-06-13	2021-10-06	56, RTS Road, Fifth Street	Jaffna	Northern	Sri Lanka	12
	16	Taylor James	Male	jamestaylor34@yahoo.com	9412395167	1999-11-08	2022-02-08	No. 56, Central Avenue	Dehiwala	Western	Sri Lanka	15

Figure 136: Integration test case 11 - Evidence #2

Staff Home > Staff Management > Manage Staff

Manage Staff

Show 10 entries Search:

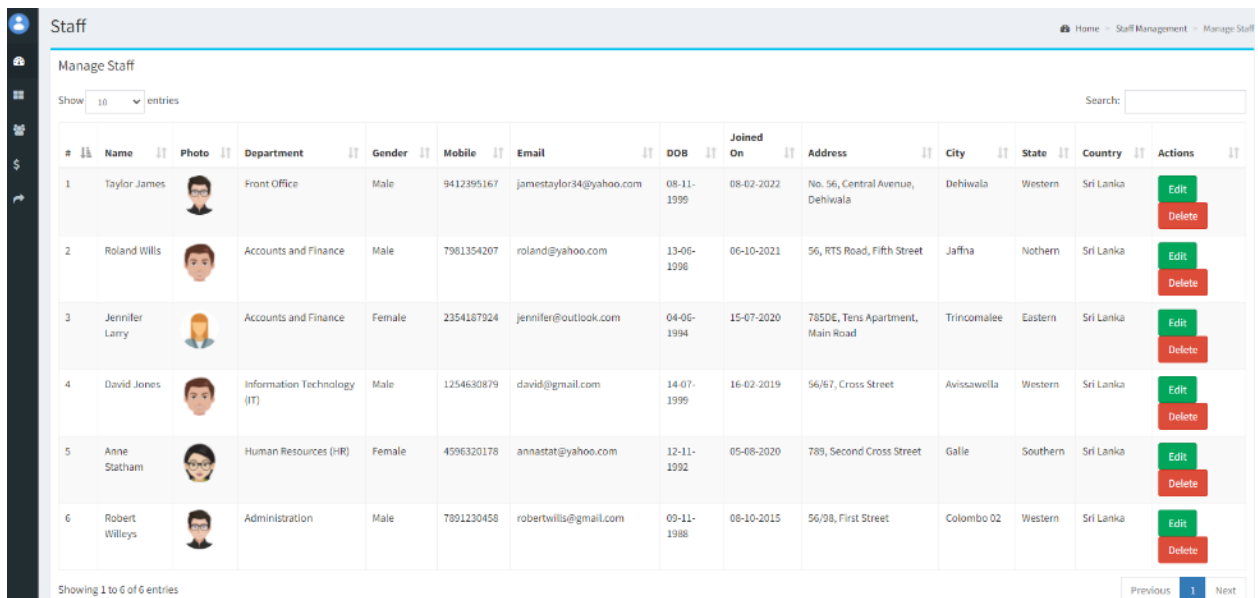
#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Taylor James		Front Office	Male	9412395167	jamestaylor34@yahoo.com	08-11-1999	08-02-2022	No. 56, Central Avenue, Dehiwala	Dehiwala	Western	Sri Lanka	
2	Roland Wills		Accounts and Finance	Male	7981354207	roland@yahoo.com	13-06-1998	06-10-2021	56, RTS Road, Fifth Street	Jaffna	Northern	Sri Lanka	
3	Jennifer Larry		Accounts and Finance	Female	2354187924	jennifer@outlook.com	04-06-1994	15-07-2020	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	

Figure 137: Integration test case 11 - Evidence #3

Table 36: Integration test case 11.1

Test Target		Add and manage employees.		
Test Description		View list of existing employees.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
11.1	-	View list of existing employees.	View list of existing employees.	PASS

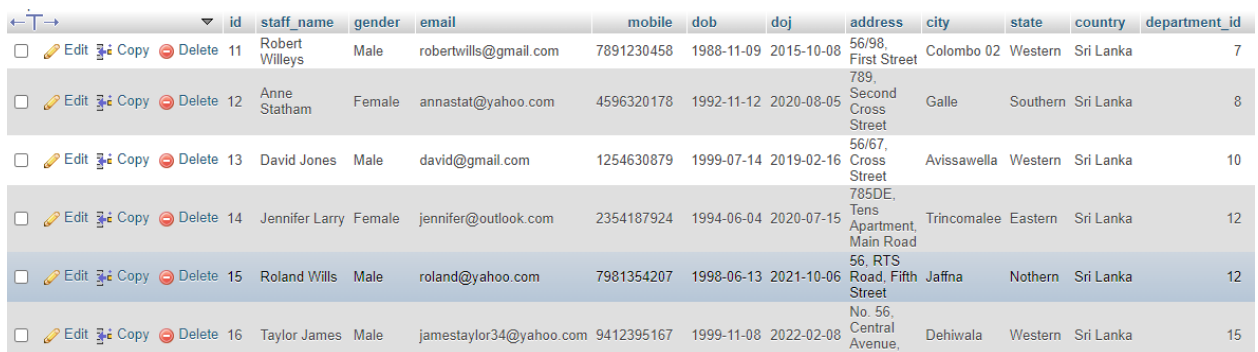
Evidence:



The screenshot shows a web application interface for managing staff. It includes a sidebar with navigation icons, a top navigation bar with 'Home', 'Staff Management', and 'Manage Staff'. The main content area is titled 'Manage Staff' and features a table with 6 entries. Each entry includes a photo, name, department, gender, mobile, email, DOB, joined date, address, city, state, country, and actions (Edit, Delete). The table is sorted by ID, and the first 6 entries are displayed.

#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Taylor James		Front Office	Male	9412395167	jamestaylor34@yahoo.com	08-11-1999	08-02-2022	No. 56, Central Avenue, Dehiwala	Dehiwala	Western	Sri Lanka	Edit, Delete
2	Roland Wills		Accounts and Finance	Male	7981354207	roland@yahoo.com	13-06-1996	06-10-2021	56, RTS Road, Fifth Street	Jaffna	Northen	Sri Lanka	Edit, Delete
3	Jennifer Larry		Accounts and Finance	Female	2354187924	jennifer@outlook.com	04-06-1994	15-07-2020	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	Edit, Delete
4	David Jones		Information Technology (IT)	Male	1254630879	david@gmail.com	14-07-1999	16-02-2019	56/67, Cross Street	Avissawella	Western	Sri Lanka	Edit, Delete
5	Anne Statham		Human Resources (HR)	Female	4596320178	annastat@yahoo.com	12-11-1992	05-08-2020	789, Second Cross Street	Galle	Southern	Sri Lanka	Edit, Delete
6	Robert Willeys		Administration	Male	7891230458	robertwills@gmail.com	09-11-1988	08-10-2015	56/98, First Street	Colombo 02	Western	Sri Lanka	Edit, Delete

Figure 138: Integration test case 11.1 - Evidence #1



The screenshot shows a web application interface for managing staff. It includes a sidebar with navigation icons, a top navigation bar with 'Home', 'Staff Management', and 'Manage Staff'. The main content area is titled 'Manage Staff' and features a table with 6 entries. Each entry includes a photo, name, department, gender, mobile, email, DOB, joined date, address, city, state, country, and actions (Edit, Delete). The table is sorted by ID, and the first 6 entries are displayed.

#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Taylor James		Front Office	Male	9412395167	jamestaylor34@yahoo.com	08-11-1999	08-02-2022	No. 56, Central Avenue, Dehiwala	Dehiwala	Western	Sri Lanka	Edit, Delete
2	Roland Wills		Accounts and Finance	Male	7981354207	roland@yahoo.com	13-06-1996	06-10-2021	56, RTS Road, Fifth Street	Jaffna	Northen	Sri Lanka	Edit, Delete
3	Jennifer Larry		Accounts and Finance	Female	2354187924	jennifer@outlook.com	04-06-1994	15-07-2020	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	Edit, Delete
4	David Jones		Information Technology (IT)	Male	1254630879	david@gmail.com	14-07-1999	16-02-2019	56/67, Cross Street	Avissawella	Western	Sri Lanka	Edit, Delete
5	Anne Statham		Human Resources (HR)	Female	4596320178	annastat@yahoo.com	12-11-1992	05-08-2020	789, Second Cross Street	Galle	Southern	Sri Lanka	Edit, Delete
6	Robert Willeys		Administration	Male	7891230458	robertwills@gmail.com	09-11-1988	08-10-2015	56/98, First Street	Colombo 02	Western	Sri Lanka	Edit, Delete

Figure 139: Integration test case 11.1 - Evidence #2

Table 37: Integration test case 11.2

Test Target		Add and manage employees.		
Test Description		Update existing employee record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
11.2	Date of Birth: 05/05/1999 Reg Date: 08/02/2022 City: Trincomalee State: Eastern Address: 45/3D, Second Cross Street	Update selected employee record.	Update selected employee record.	PASS

Evidence:

Staff Management

Edit Staff

Full Name: Taylor James

Gender: Male

Mobile: 9412395167

Date of Birth: 05/05/1999

City: Trincomalee

Country: Sri Lanka

Department: Front Office

Email: jamestaylor34@yahoo.com

Photo: Choose File

Date of Joining: 08/02/2022

State: Eastern

Address: 45/3D, Second Cross Street

Submit

Figure 140: Integration test case 11.2 - Evidence #1

Manage Staff

Show 10 entries

Search:

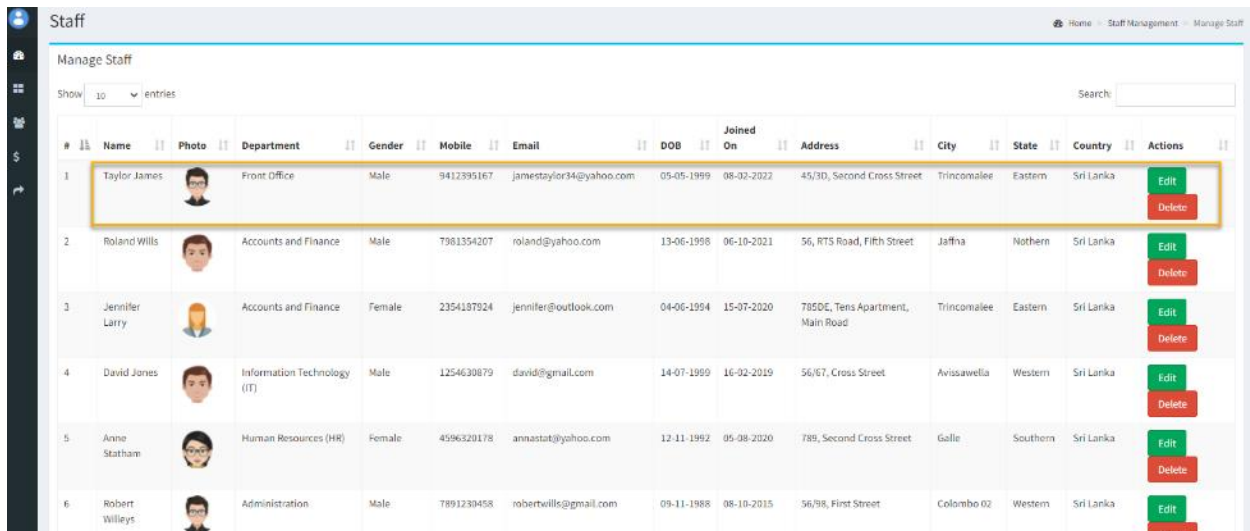
#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Taylor James		Front Office	Male	9412395167	jamestaylor34@yahoo.com	05-05-1999	08-02-2022	45/3D, Second Cross Street	Trincomalee	Eastern	Sri Lanka	Edit Delete

Figure 141: Integration test case 11.2 - Evidence #2

Table 38: Integration test case 11.3

Test Target		Add and manage employees.		
Test Description		Delete existing employee record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
11.3	-	Delete selected employee record.	Delete selected employee record.	PASS

Evidence:









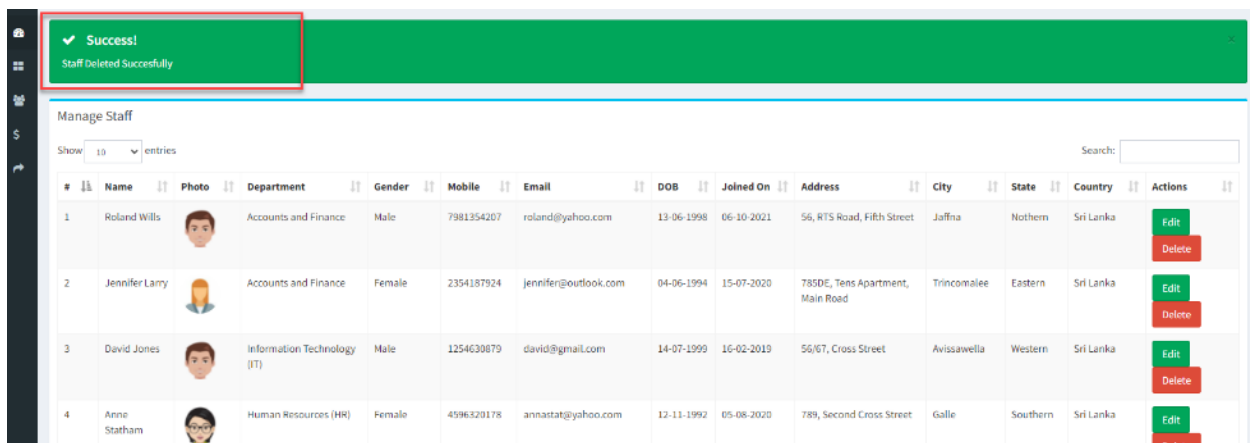
#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Taylor James		Front Office	Male	9412395167	jamestaylor34@yahoo.com	05-05-1999	08-02-2022	45/3D, Second Cross Street	Trincomalee	Eastern	Sri Lanka	Edit Delete
2	Roland Wills		Accounts and Finance	Male	7981354207	roland@yahoo.com	13-06-1998	06-10-2021	56, RTS Road, Fifth Street	Jaffna	Nothorn	Sri Lanka	Edit Delete
3	Jennifer Larry		Accounts and Finance	Female	2354187924	jennifer@outlook.com	04-06-1994	15-07-2020	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	Edit Delete
4	David Jones		Information Technology (IT)	Male	1254630879	david@gmail.com	14-07-1999	16-02-2019	56/67, Cross Street	Avisawella	Western	Sri Lanka	Edit Delete
5	Anne Statham		Human Resources (HR)	Female	4596320178	annastat@yahoo.com	12-11-1992	05-08-2020	789, Second Cross Street	Galle	Southern	Sri Lanka	Edit Delete
6	Robert Willets		Administration	Male	7891230458	robertwills@gmail.com	09-11-1988	08-10-2015	56/98, First Street	Colombo 02	Western	Sri Lanka	Edit Delete

Figure 142: Integration test case 11.3 - Evidence #1







#	Name	Photo	Department	Gender	Mobile	Email	DOB	Joined On	Address	City	State	Country	Actions
1	Roland Wills		Accounts and Finance	Male	7981354207	roland@yahoo.com	13-06-1998	06-10-2021	56, RTS Road, Fifth Street	Jaffna	Nothorn	Sri Lanka	Edit Delete
2	Jennifer Larry		Accounts and Finance	Female	2354187924	jennifer@outlook.com	04-06-1994	15-07-2020	785DE, Tens Apartment, Main Road	Trincomalee	Eastern	Sri Lanka	Edit Delete
3	David Jones		Information Technology (IT)	Male	1254630879	david@gmail.com	14-07-1999	16-02-2019	56/67, Cross Street	Avisawella	Western	Sri Lanka	Edit Delete
4	Anne Statham		Human Resources (HR)	Female	4596320178	annastat@yahoo.com	12-11-1992	05-08-2020	789, Second Cross Street	Galle	Southern	Sri Lanka	Edit Delete

Figure 143: Integration test case 11.3 - Evidence #2

Table 39: Integration test case 12

Test Target		Add and manage employee payroll.		
Test Description		Add new employee payroll.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
12	Name: Anne Statham Basic Salary: 98000 Allowance: 8450	Department added as database record.	Department added as database record.	PASS

Evidence:

Salary

Home > Salary > Add Salary

Add Employee Salary

Department Name
Human Resources (HR)

Staff	Basic Salary (LKR)	Allowance (LKR)	Total (LKR)
Anne Statham	98000	8450	106450

Submit

Figure 144: Integration test case 12 - Evidence #1

Salary Management

Home

Manage Employee Payrolls

Show 10 entries

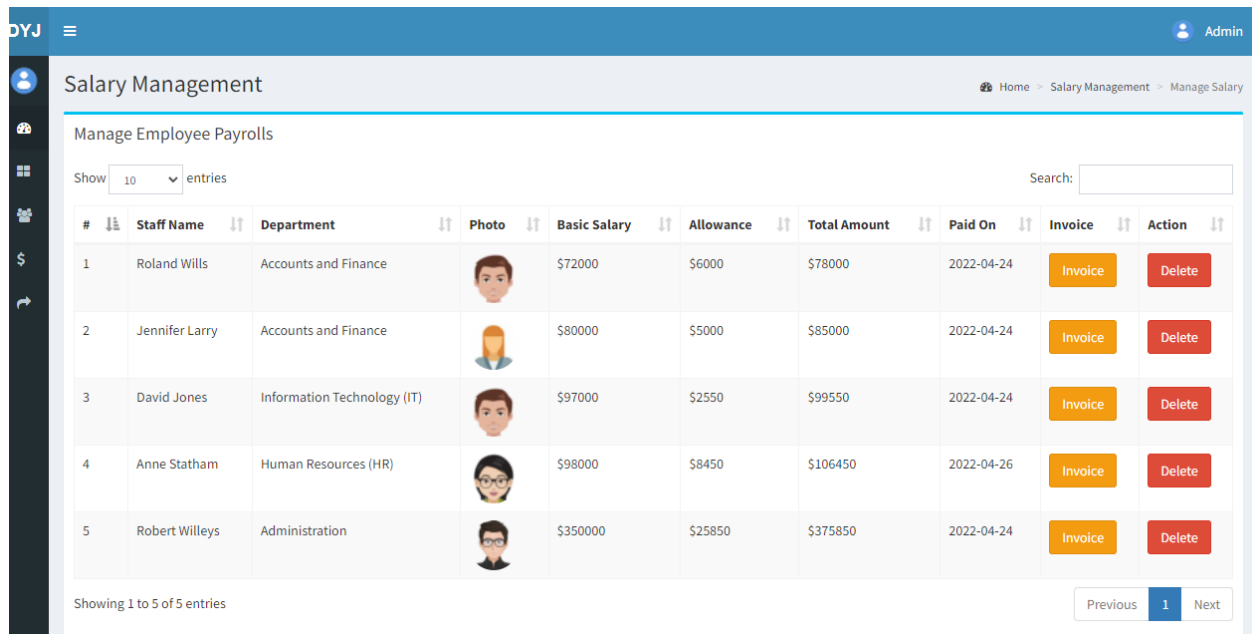
#	Staff Name	Department	Photo	Basic Salary	Allowance	Total Amount	Paid On
4	Anne Statham	Human Resources (HR)		\$98000	\$8450	\$106450	2022-04-26

Figure 145: Integration test case 12 - Evidence #1

Table 40: Integration test case 12.1

Test Target		Add and manage employee payroll.		
Test Description		View employee payroll history.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
12.1	-	View list of employee payroll history.	View list of employee payroll history.	PASS

Evidence:



The screenshot shows a web application interface for 'Salary Management'. The main heading is 'Manage Employee Payrolls'. Below this, there is a search bar and a table with 5 entries. The table columns are: #, Staff Name, Department, Photo, Basic Salary, Allowance, Total Amount, Paid On, Invoice, and Action. The entries are for Roland Wills, Jennifer Larry, David Jones, Anne Statham, and Robert Willeys. Each entry has an 'Invoice' button and a 'Delete' button. The interface also shows a sidebar with navigation icons and a top bar with the user 'Admin'.

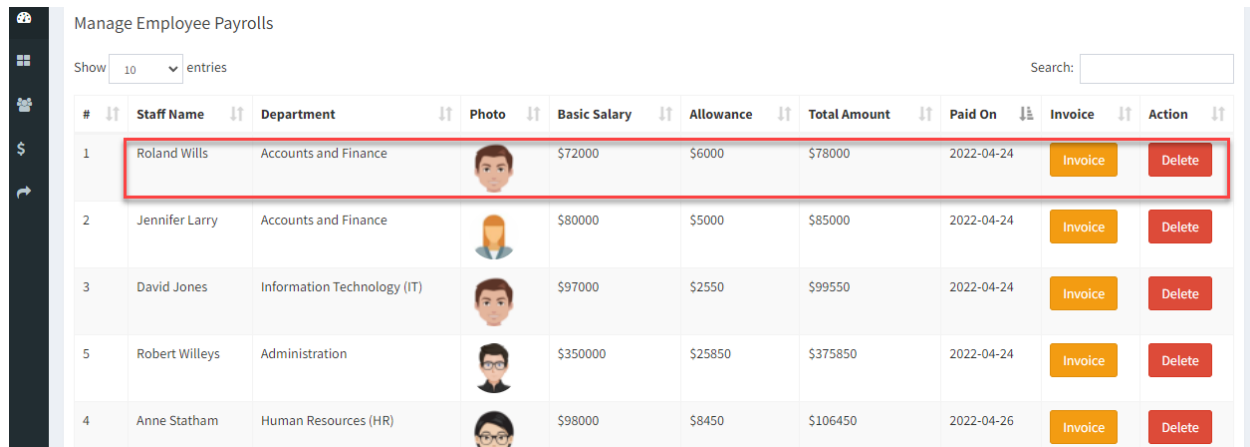
#	Staff Name	Department	Photo	Basic Salary	Allowance	Total Amount	Paid On	Invoice	Action
1	Roland Wills	Accounts and Finance		\$72000	\$6000	\$78000	2022-04-24	Invoice	Delete
2	Jennifer Larry	Accounts and Finance		\$80000	\$5000	\$85000	2022-04-24	Invoice	Delete
3	David Jones	Information Technology (IT)		\$97000	\$2550	\$99550	2022-04-24	Invoice	Delete
4	Anne Statham	Human Resources (HR)		\$98000	\$8450	\$106450	2022-04-26	Invoice	Delete
5	Robert Willeys	Administration		\$350000	\$25850	\$375850	2022-04-24	Invoice	Delete

Figure 146: Integration test case 12.1 - Evidence #1

Table 41: Integration test case 12.2

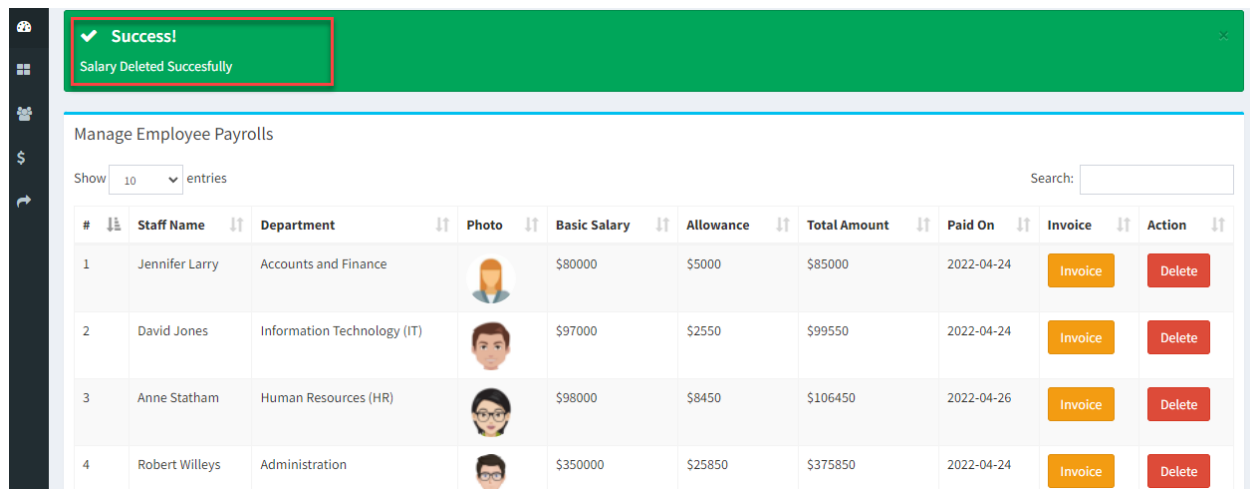
Test Target		Add and manage employee payroll.		
Test Description		Delete employee payroll record.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
12.2	-	Delete selected employee payroll record.	Delete selected employee payroll record.	PASS

Evidence:



#	Staff Name	Department	Photo	Basic Salary	Allowance	Total Amount	Paid On	Invoice	Action
1	Roland Wills	Accounts and Finance		\$72000	\$6000	\$78000	2022-04-24	Invoice	Delete
2	Jennifer Larry	Accounts and Finance		\$80000	\$5000	\$85000	2022-04-24	Invoice	Delete
3	David Jones	Information Technology (IT)		\$97000	\$2550	\$99550	2022-04-24	Invoice	Delete
5	Robert Willeys	Administration		\$350000	\$25850	\$375850	2022-04-24	Invoice	Delete
4	Anne Statham	Human Resources (HR)		\$98000	\$8450	\$106450	2022-04-26	Invoice	Delete

Figure 147: Integration test case 12.2 - Evidence #1



#	Staff Name	Department	Photo	Basic Salary	Allowance	Total Amount	Paid On	Invoice	Action
1	Jennifer Larry	Accounts and Finance		\$80000	\$5000	\$85000	2022-04-24	Invoice	Delete
2	David Jones	Information Technology (IT)		\$97000	\$2550	\$99550	2022-04-24	Invoice	Delete
3	Anne Statham	Human Resources (HR)		\$98000	\$8450	\$106450	2022-04-26	Invoice	Delete
4	Robert Willeys	Administration		\$350000	\$25850	\$375850	2022-04-24	Invoice	Delete

Figure 148: Integration test case 12.2 - Evidence #2

Table 42: Integration test case 13

Test Target		Generate employee salary invoice.		
Test Description		Invoice generated with employee and salary information.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
13	-	Salary invoice generated with relevant details.	Salary invoice generated with relevant details.	PASS

Evidence:

Salary Invoice #00349 Home > Salary Management > Invoice

Don Y.J Holdings | Centralized Management System Date: 26-04-2022


From:
Financial Controller
Don Y.J Holdings (Pvt) Ltd
No. 17, Hewa Avenue,
Colombo 07, 00700
Sri Lanka
Telephone: 011 271 2713
Email: donyj@palmbeach.lk

To:
Anne Statham
Galle
Southern, Sri Lanka
Mobile: 4586320178
Email: annestat@yahoo.com

Invoice #0011
Paid On: 26-04-2022

#	Basic Salary	Allowance	Subtotal
1	LKR 98000/=	LKR 8450/=	LKR 106450/=

Payment Method:



Salary Details

Subtotal:	LKR 106450/=
Tax (0%):	LKR 0/=
Total:	LKR 106450/=

Dear Anne Statham, your employee salary has been transferred to your respective bank account. Your payment has been confirmed by the Financial Controller of Don Y.J Holdings, on 26-04-2022.
Please contact the Financial Controller if you haven't received the payment. © Don Y.J Holdings.

[Print](#)
[Generate PDF](#)

Figure 149: Integration test case 13 - Evidence #1

Table 43: Integration test case 14

Test Target		Create employee leave requests.		
Test Description		Request for employee leave based on required dates.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
14	-	Add employee leave request record.	Add employee leave request record.	PASS

Evidence:

The screenshot shows the 'Apply Leave' form in a web application. The form has a blue header bar with a menu icon and the text 'Staff'. Below the header, the page title is 'Leave Management' and the breadcrumb is 'Home > Leave Management > Apply Leave'. The form itself is titled 'Apply Leave' and contains four main input sections: 'Reason' with a text box containing 'High fever', 'Leave From' with a date picker showing '27/04/2022', 'Leave To' with a date picker showing '01/05/2022', and 'Description' with a text box containing 'Sick leave'. A green 'Submit' button is located at the bottom right of the form.

Figure 150: Integration test case 14 - Evidence #1

The screenshot shows the 'View Leave' table in a web application. The page title is 'Leave Management' and the breadcrumb is 'Home > Leave Management > View Leave'. The table has columns: '#', 'Reason', 'From', 'To', 'Status', 'Description', and 'Applied On'. There is one entry in the table, which is highlighted with a red border. The entry details are: # 1, Reason High fever, From 27-04-2022, To 01-05-2022, Status Pending, Description Sick leave, and Applied On 26-04-2022. The table also includes a search bar, a 'Show 10 entries' dropdown, and pagination controls at the bottom.

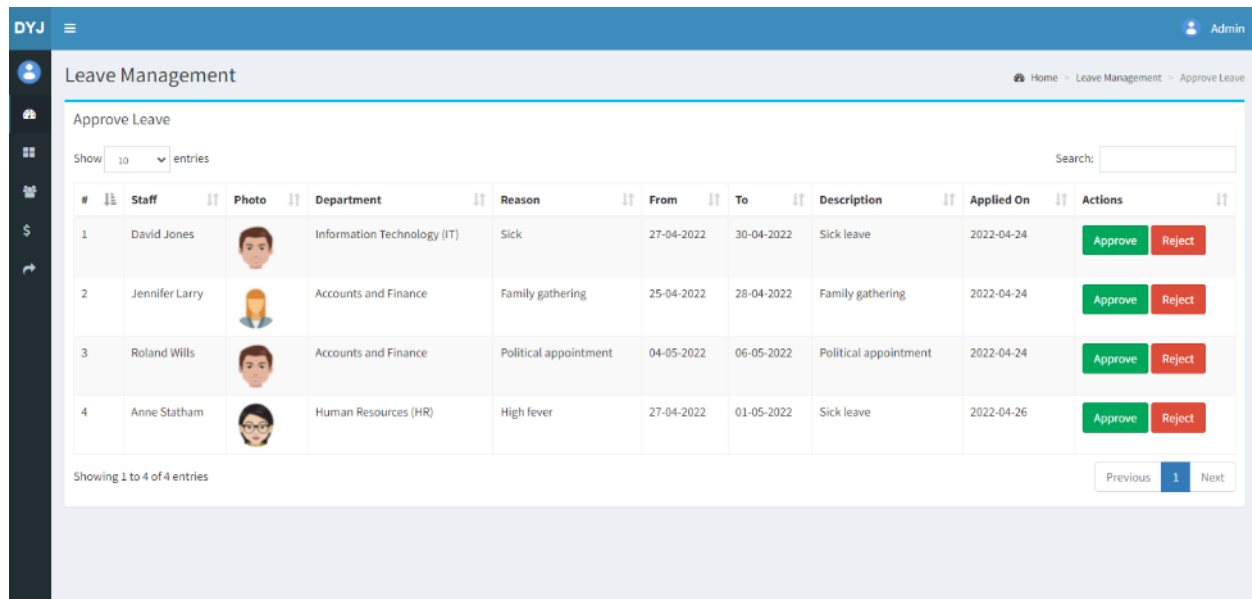
#	Reason	From	To	Status	Description	Applied On
1	High fever	27-04-2022	01-05-2022	Pending	Sick leave	26-04-2022

Figure 151: Integration test case 14 - Evidence #1

Table 44: Integration test case 15

Test Target		View and manage employee leave requests.		
Test Description		View existing employees leave requests.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
15	-	View list of employees leave requests.	View list of employees leave requests.	PASS

Evidence:







#	Staff	Photo	Department	Reason	From	To	Description	Applied On	Actions
1	David Jones		Information Technology (IT)	Sick	27-04-2022	30-04-2022	Sick leave	2022-04-24	<button>Approve</button> <button>Reject</button>
2	Jennifer Larry		Accounts and Finance	Family gathering	25-04-2022	28-04-2022	Family gathering	2022-04-24	<button>Approve</button> <button>Reject</button>
3	Roland Wills		Accounts and Finance	Political appointment	04-05-2022	06-05-2022	Political appointment	2022-04-24	<button>Approve</button> <button>Reject</button>
4	Anne Statham		Human Resources (HR)	High fever	27-04-2022	01-05-2022	Sick leave	2022-04-26	<button>Approve</button> <button>Reject</button>

Figure 152: Integration test case 15 - Evidence #1

Table 45: Integration test case 15.1

Test Target		View and manage employee leave requests.		
Test Description		Update status of existing leave requests.		
Test Case	Input Data	Expected Outcome	Actual Outcome	Result (Pass/Fail)
15.1	“Approve”	Change leave status to “Approved”	Change leave status to “Approved”	PASS

Evidence:

Approve Leave





Show entries Search:

#	Staff	Photo	Department	Reason	From	To	Description	Applied On	Actions
1	David Jones		Information Technology (IT)	Sick	27-04-2022	30-04-2022	Sick leave	2022-04-24	<button>Approve</button> <button>Reject</button>
2	Jennifer Larry		Accounts and Finance	Family gathering	25-04-2022	28-04-2022	Family gathering	2022-04-24	<button>Approve</button> <button>Reject</button>
3	Roland Wills		Accounts and Finance	Political appointment	04-05-2022	06-05-2022	Political appointment	2022-04-24	<button>Approve</button> <button>Reject</button>
4	Anne Statham		Human Resources (HR)	High fever	27-04-2022	01-05-2022	Sick leave	2022-04-26	<button>Approve</button> <button>Reject</button>

Figure 153: Integration test case 15.1 - Evidence #1

View Leave

Show entries Search:

#	Staff	Photo	Department	Reason	From	To	Status	Description	Applied On
4	David Jones		Information Technology (IT)	Sick	27-04-2022	30-04-2022	Approved	Sick leave	2022-04-24
1	Anne Statham		Human Resources (HR)	High fever	27-04-2022	01-05-2022	Pending	Sick leave	2022-04-26
2	Roland Wills		Accounts and Finance	Political appointment	04-05-2022	06-05-2022	Pending	Political appointment	2022-04-24
3	Jennifer Larry		Accounts and Finance	Family gathering	25-04-2022	28-04-2022	Pending	Family gathering	2022-04-24

Showing 1 to 4 of 4 entries Previous **1** Next

Figure 154: Integration test case 15.1 - Evidence #2

Integration Testing Report:

Table 46: Integration testing report

Integration Testing Report Don Y.J Holdings (Pvt) Ltd Online Reservation and Centralized Management System			
Tester: Ravindradaas Kogul Sager			
Test Case ID	Test Target	Test Date	Result
01	Room reservation.	21/04/2022	PASS
02	Guest user registration.	10/02/2022	PASS
03	Guest user login.	10/02/2022	PASS
04	Reservation system administrator login.	22/04/2022	PASS
05	Add and manage room types.	23/04/2022	PASS
06	Add and manage rooms.	24/04/2022	PASS
07	View and manage reservations.	22/04/2022	PASS
08	Generate and print booking report.	25/04/2022	PASS

09	Management system login.	23/04/2022	PASS
10	Add and manage departments.	21/04/2022	PASS
11	Add and manage employees.	25/04/2022	PASS
12	Add and manage employee payroll.	26/04/2022	PASS
13	Generate employee salary invoice.	26/04/2022	PASS
14	Create employee leave requests.	26/04/2022	PASS
15	View and manage employee leave requests.	26/04/2022	PASS

Non-Functional Testing:

Non-functional testing is the testing procedure of verifying the software application's non-function aspects including its performance, usability and functional reliability. The testing validates the effectiveness of the application based on its non-functional variables which would not be included in functional testing. Basically, non-functional testing would verify the application on how user-friendly the system would be in the hands of the real-world users. Moreover, factors such as the application's maintenance ability could also be tested. In summary, non-functional testing helps to eliminate potential user experience risks. (Hamilton, 2021)

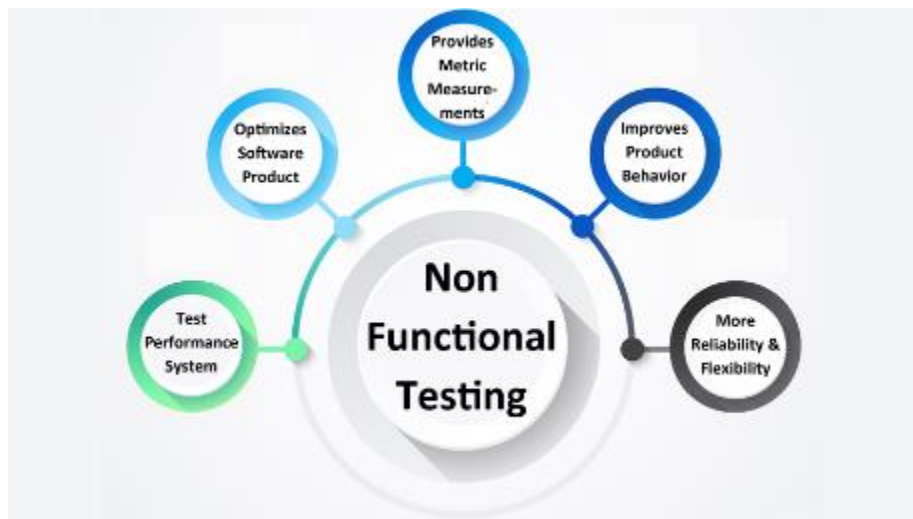


Figure 155: Non-Functional Testing

The diagram below illustrates the procedure of a typical non-functional testing;

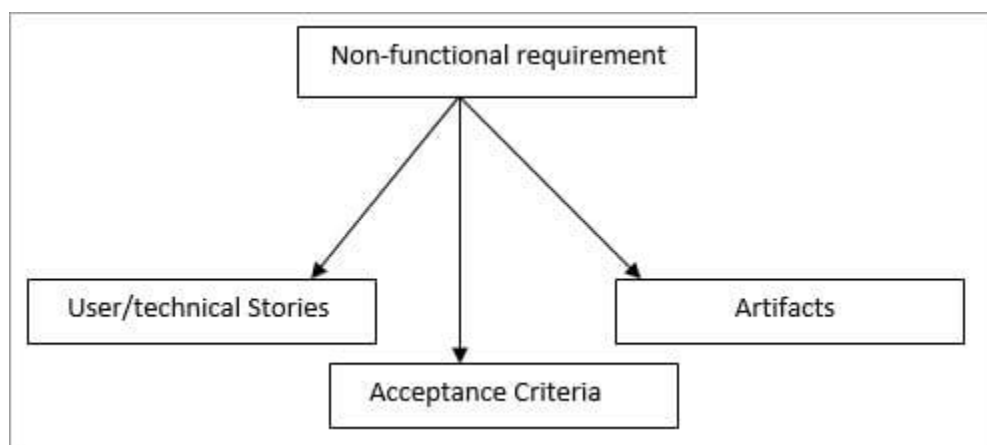







Figure 156: Non-functional testing procedure illustration

Feedback:

Considering the progress of the current project, a feedback form was provided to an administrative staff member of Don Y.J Holdings (Pvt) Ltd, requesting valuable feedback on the current project progress. The filled feedback form is shown below.

Don Y.J Holdings
Progress Feedback Form
Online Reservation and Centralized Management System

					
	Very Poor	Poor	Satisfactory	Very Good	Excellent
What was your first impression on the project progress?					✓
How would you rate the requirement analysis phase?					✓
How would you rate the system development approaches?				✓	
How would you rate on the security of the system?				✓	
How would you rate the standards of the project?					✓
How would you rate the system functional approaches?				✓	
How would you rate the system user interface designs?					✓
Additional Feedback:	Good Progress on the project. Looks professional More creativity expected. Good Luck.				

R. S. R. Das [Signature] 14th February 2022
Name Signature Date
Don Y.J Holdings (Pvt) Limited
20, 27, New Avenue
Colombo - 07.

Figure 157: Client administrative staff feedback form

Visual feedback from the employees of the client company was collected. A “Google Form” was created and shared among ten company employees. The form results are explained below.

The first question was asking if the employees think that the company required an automated management system to carry on their day-to-day tasks. 100% of the employees have agreed that the company currently requires an automated management system.

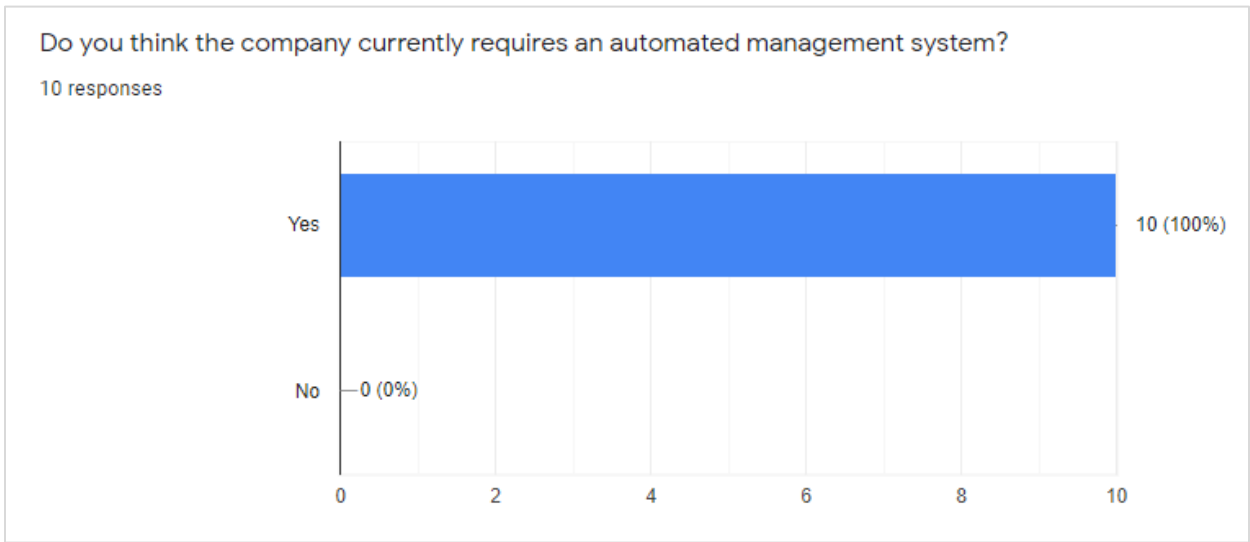


Figure 158: Employee feedback form result - question 01

The second question was the employee’s initial impression towards the system development. 40% of the employees have rated a 5/5, with another 40% rating 4/5, while 20% of the employees have rated 3/5 as the project’s initial impression.

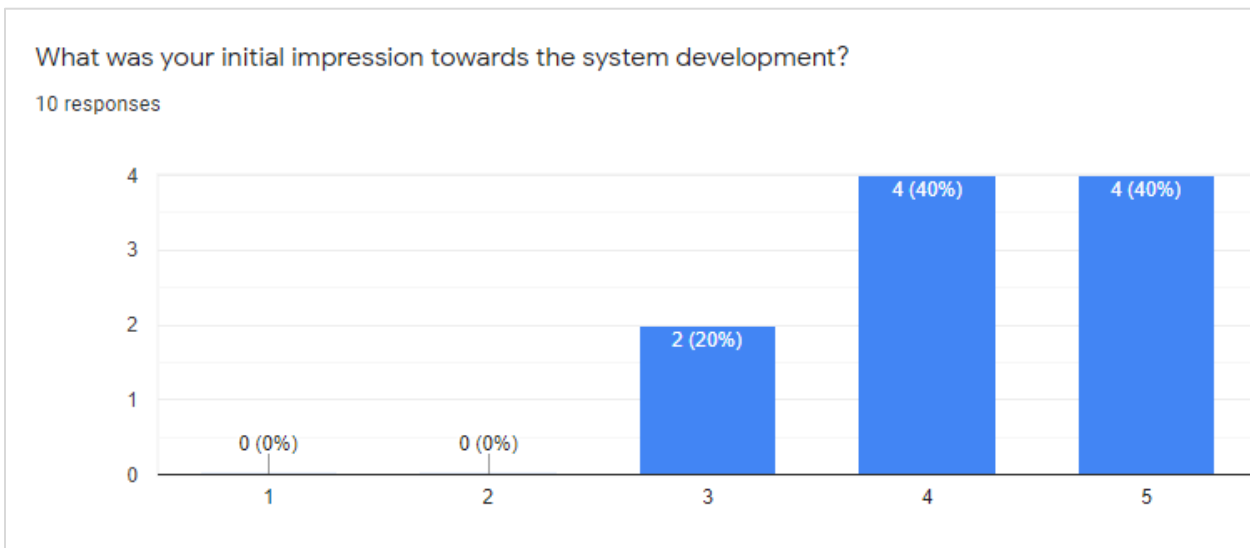


Figure 159: Employee feedback form result - question 02

The third question was to rate the analysis phase of the project. 40% of the employees have rated a 5/5, with another 50% rating 4/5, while 10% of the employees have rated 3/5. This represents that the project has showcased some satisfying system analysis phase.

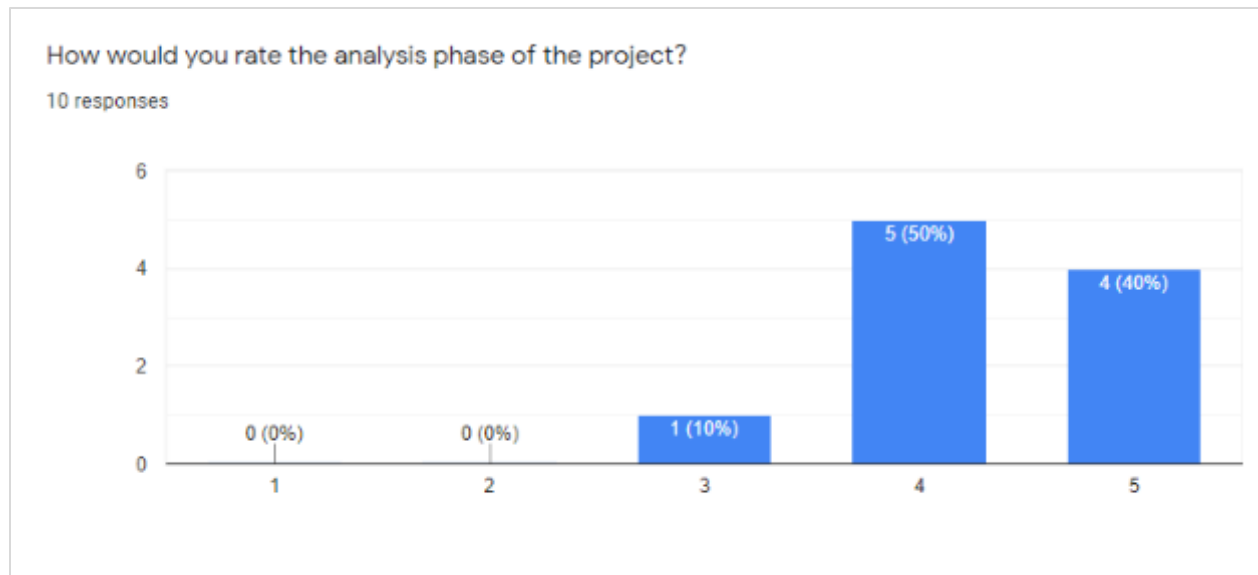


Figure 160: Employee feedback form result - question 03

The fourth question was on how satisfied the employees were with the system development approached of the application development. 40% of the employees have rated a 5/5, with another 40% rating 4/5, while 20% of the employees have rated 3/5.

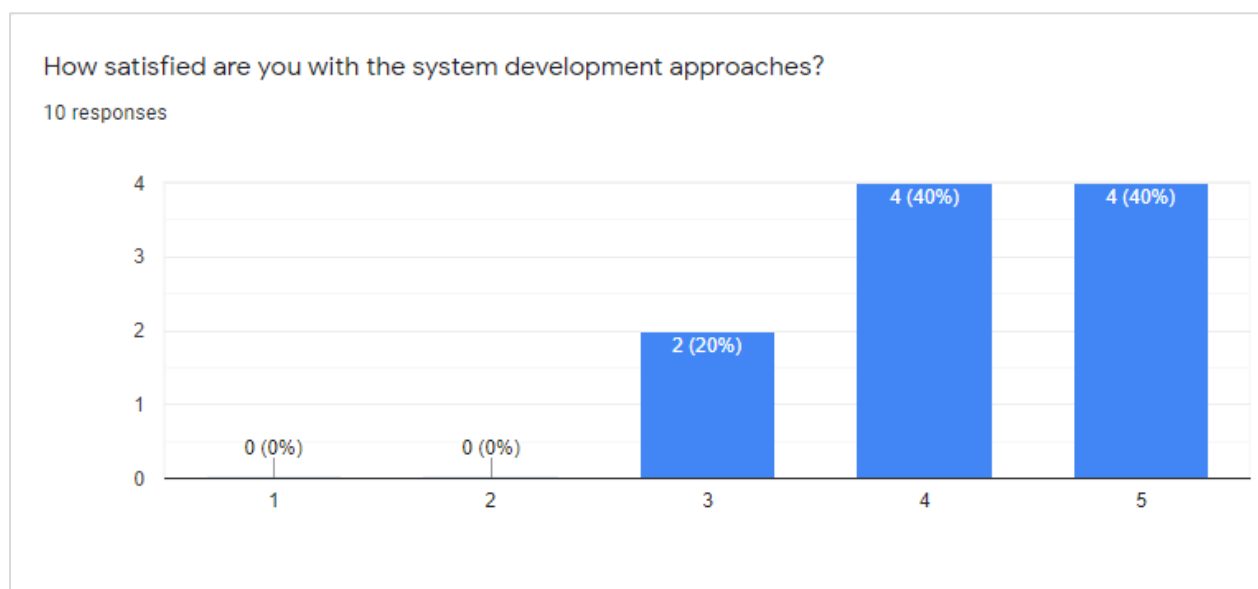


Figure 161: Employee feedback form result - question 04

The fifth question was if the employees found the application requirements to be properly identified and categorized in the project. 100% of the employees have found this to be true, which indicates that the requirements gathering phase of the project has been successful.

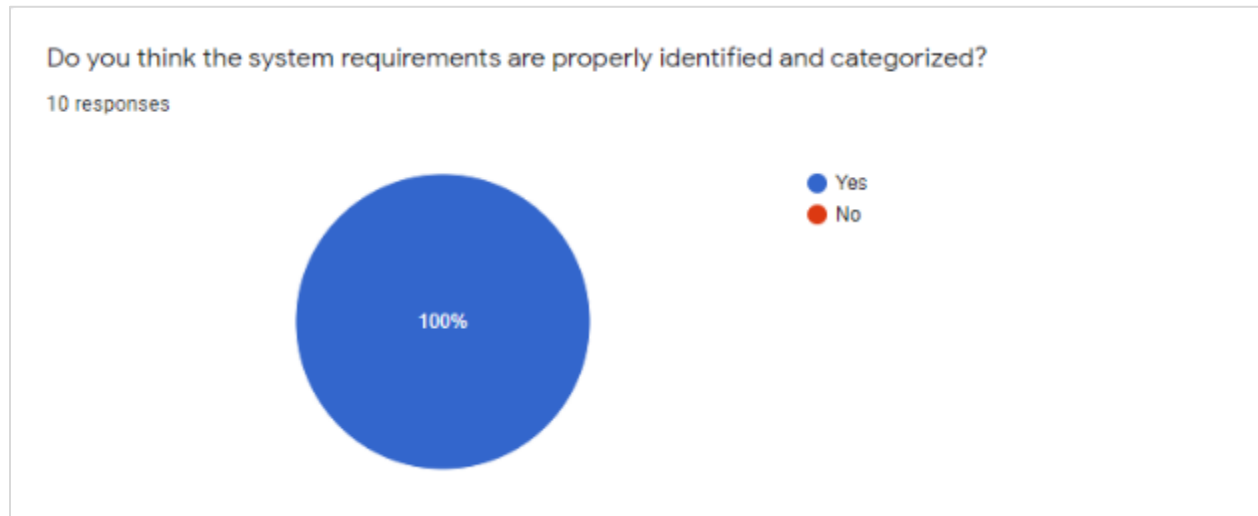


Figure 162: Employee feedback form result - question 05

The sixth question was to rate the front-end user interface of the project web application system. 50% of the employees have rated a 5/5, with another 30% rating 4/5, while 20% of the employees have rated 3/5 for the user interface designs.

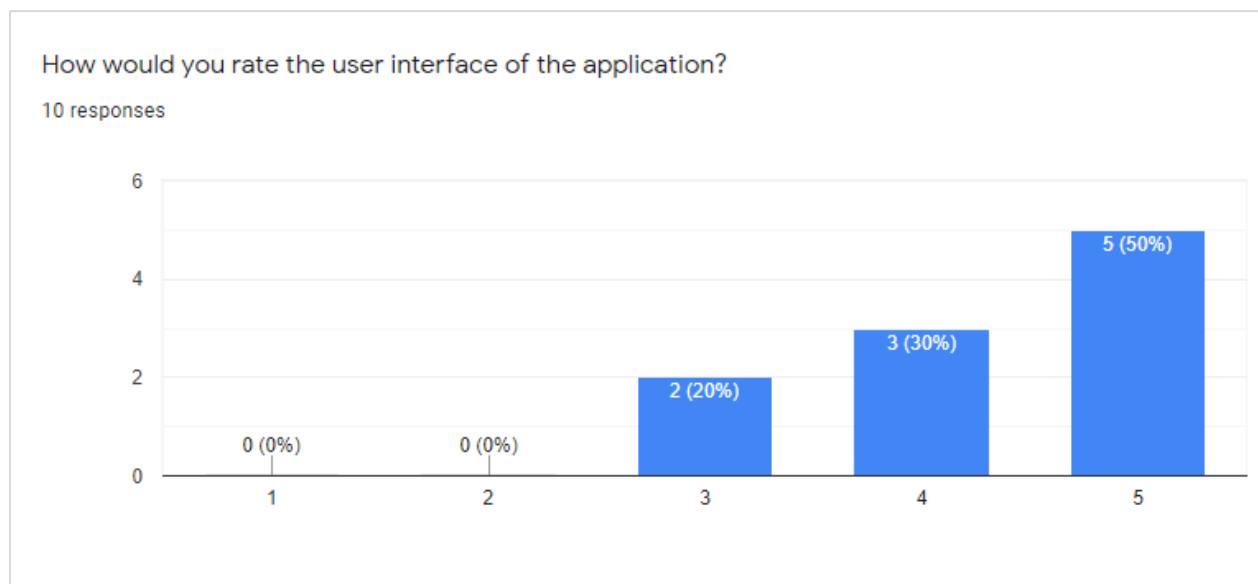
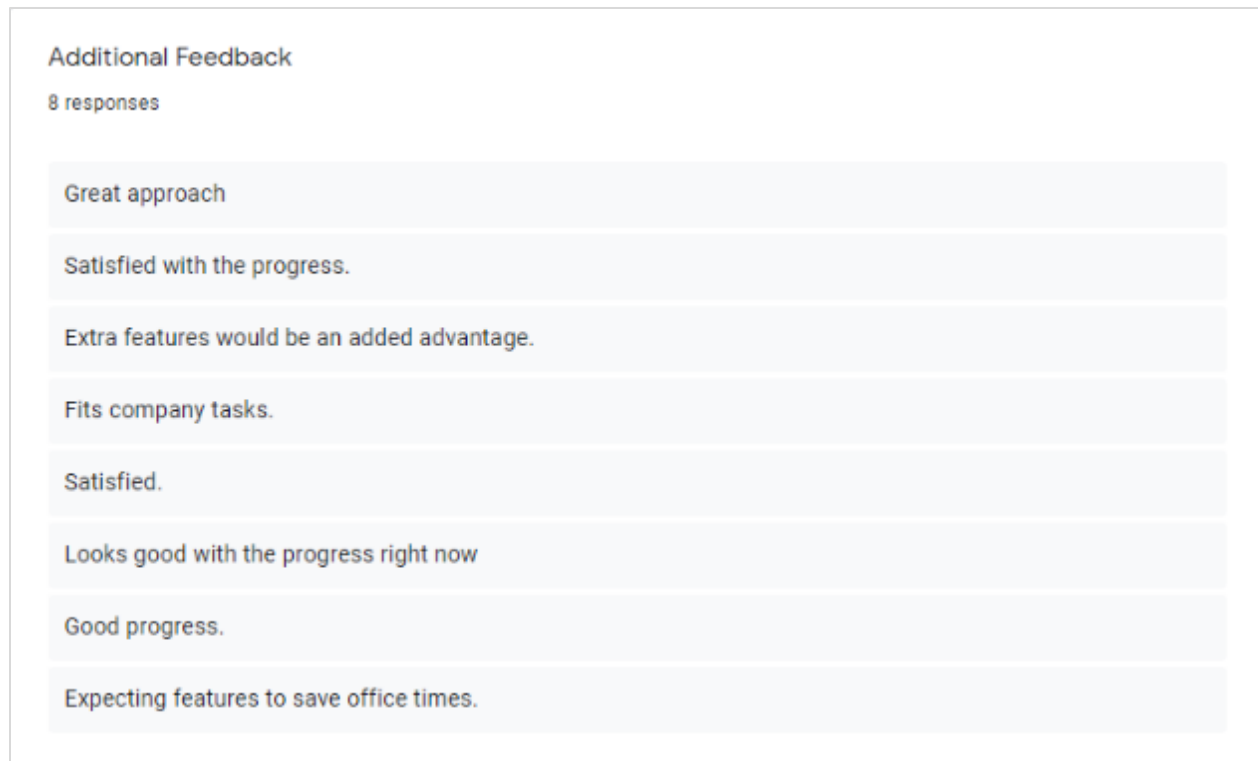


Figure 163: Employee feedback form result - question 06

The final question of the feedback form was dedicated as an additional feedback form for employees to express their impressions and suggest further ideas for the project. Answers from six employees were collected for this question, as shown below;



The screenshot displays a feedback form titled 'Additional Feedback' with a subtitle '8 responses'. Below the title, there are eight text boxes, each containing a response from an employee. The responses are: 'Great approach', 'Satisfied with the progress.', 'Extra features would be an added advantage.', 'Fits company tasks.', 'Satisfied.', 'Looks good with the progress right now', 'Good progress.', and 'Expecting features to save office times.'

Response
Great approach
Satisfied with the progress.
Extra features would be an added advantage.
Fits company tasks.
Satisfied.
Looks good with the progress right now
Good progress.
Expecting features to save office times.

Figure 164: Employee feedback form result - question 07

Feedback Evaluation:

Considering the feedback towards the project from Don Y.J Holdings company, it can be concluded that the project phase is moving towards the correct decision, in consideration with the currently evaluated system analysis, research and system requirements. The client feedback has been advantageous to make sure that the intended solution actually fits with the initial problem. This feedback collection would be an essential section of the software quality assurance to make sure proper analysis, research and evaluation has been intended in the project.

6. Critical Review & Conclusion

6.1 Closing Executive Summary

An effective analysis and evaluation for the project progress of developing an online reservation and centralized management system for Don Y.J Holdings has assisted to understand the proper project flow. The project targets to develop an online hotel reservation system for the company, with a centralized employee/client management system to store/access company data remotely. The management system will also include a business accounts management system, which the company authorities and employees can use to report and manage the company's business and account records digitally, which can be accessed remotely. This analysis has revealed proper project scopes along with the knowledge of similar projects. The evaluations of the relevant analysis have assisted the project to make sure the process ensures that the company's relevant requirements are being properly considered for the development of the application.

The project procedures should be conducted in a secure and company-authorized manner. It is also essential that the application is included with proper data protection elements, to prevent data loss or theft. The system would also ensure user permissions so that different users can access only the data they are authorized with. Although the scope of the project is currently straightforward, certain approaches could have been directed in consideration of the limited time available for the project. The initial objectives of the project are already taking up a longer time period of the entire project period. A minor scope adjustment of the project would have been to reduce the analysis procedure times in order to allow more time for application development.

6.2 Conclusion

The final aim of this project is to develop an online reservation and centralized management system with the use of web application technology for Don Y.J Holdings, to solve the problems faced by the company based on information gathering and business management. The holdings company contains of three tourist hotels and two bungalows. Currently, each hotel/bungalow has a separate system to store/access employee/client information. The company requires a dedicated reservation system for users to reserve hotels and bungalows. The outcome of the project would solve all problems faced by the company currently, as well as offer newer business opportunities.

The major strength of the current project application is that the approach to solve the company's problem is straightforward and used efficient techniques for the solving procedure. The application is planned with the thought of directly solving each problem with the company's current system. This ensures that every development procedure and technology is dedicated to solve the issue entirely. The application also used up-to-date latest software development tools and technologies. This makes the system ready to face any third-party competition that might exist in the relevant business market, once the application is released for customers.

Considering future improvements for the proposed application, the company would be able to expand the available functions in the application in order to automate additional business operations. The current project application is designed in a manner to support further updates in the functionalities of the system. Once the company identifies a new functional opportunity, it can be implemented and properly added to the existing system. Moreover, there is also a possibility for the company to merge with third-party business sites to expand their current holdings company. The company could also consider widening their platforms to mobile platform devices. This would also allow the business to reach among a wider and newer range of customers that the company can attract and expand their growth.

References / Bibliography

- Akazue, M. I., 2016. *Enhanced Hotel Management Information System for Multiple Reservation*, Delta State, Nigeria: s.n.
- Costa, R., 2019. *Beginners' guide to wireframing UIs: benefits & best practices*. [Online]
Available at: <https://www.justinmind.com/blog/why-wireframing-your-interface/#:~:text=A%20wireframe%20is%20often%20described,getting%20into%20the%20visual%20design>.
[Accessed 10 February 2022].
- Delizo, G. A. & Esguerra, M. A., 2013. ONLINE HOTEL RESERVATION AND MANAGEMENT SYSTEM FOR THE COLLEGE OF INTERNATIONAL TOURISM AND HOSPITALITY MANAGEMENT (CITHM). *INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY*, 10(01), p. 29.
- Hamilton, T., 2021. *Agile Methodology: What is Agile Software Development Model & Process in Testing?*. [Online]
Available at: <https://www.guru99.com/agile-scrum-extreme-testing.html>
[Accessed 08 February 2022].
- Hamilton, T., 2021. *Unit Testing Tutorial: What is, Types, Tools & Test EXAMPLE*. [Online]
Available at: <https://www.guru99.com/unit-testing-guide.html>
[Accessed 10 February 2022].
- Hamilton, T., 2021. *What is BLACK Box Testing? Techniques, Example & Types*. [Online]
Available at: <https://www.guru99.com/black-box-testing.html>
[Accessed 10 February 2022].
- Hamilton, T., 2021. *What is Non Functional Testing?*. [Online]
Available at: <https://www.guru99.com/non-functional-testing.html#:~:text=NON%2DFUNCTIONAL%20TESTING%20is%20defined,never%20addressed%20by%20functional%20testing>.
[Accessed 08 February 2022].
- Hamilton, T., 2021. *What is WHITE Box Testing? Techniques, Example & Types*. [Online]
Available at: <https://www.guru99.com/white-box-testing.html>
[Accessed 10 February 2022].
- JavaTpoint, 2021. *Integration testing*. [Online]
Available at: <https://www.javatpoint.com/integration-testing>
[Accessed 10 February 2022].
- Lacalle, E., 2021. *Top 10 benefits and advantages of a hotel management system*. [Online]
Available at: <https://www.mews.com/en/blog/advantages-of-hotel-management-system>
[Accessed 08 February 2022].
- Lutkevich, B., 2021. *What is Scrum?*. [Online]
Available at: <https://searchsoftwarequality.techtarget.com/definition/Scrum>
[Accessed 08 February 2022].
- Priyadharshini, S. & Joy, R. C., 2021. *Design and Implementation of an Automated Hotel Management System*, Coimbatore, India: Blue Eyes Intelligence Engineering.

ProfessionalQA, 2020. *Software Validation*. [Online]
 Available at: <https://www.professionalqa.com/software-validation#:~:text=Software%20Validation%20is%20a%20process,%2Fcustomers'%20demands%20and%20expectations.>
 [Accessed 08 February 2022].

Rouse, M., 2006. *PHP*. [Online]
 Available at: <https://whatis.techtarget.com/definition/PHP-Hypertext-Preprocessor>
 [Accessed 08 February 2022].

Rouse, M., 2017. *Bootstrap*. [Online]
 Available at: <https://whatis.techtarget.com/definition/bootstrap>
 [Accessed 07 February 2022].

Rouse, M., 2018. *CSS (Cascading Style Sheets)*. [Online]
 Available at: <https://www.theserverside.com/definition/cascading-style-sheet-CSS>
 [Accessed 08 February 2022].

Rouse, M., 2018. *JavaScript (JS)*. [Online]
 Available at: <https://www.theserverside.com/definition/JavaScript>
 [Accessed 08 February 2022].

Rouse, M., 2019. *Web Application*. [Online]
 Available at: [https://searchsoftwarequality.techtarget.com/definition/Web-application-Web-app#:~:text=A%20Web%20application%20\(Web%20app.all%2C%20websites%20contain%20Web%20apps.](https://searchsoftwarequality.techtarget.com/definition/Web-application-Web-app#:~:text=A%20Web%20application%20(Web%20app.all%2C%20websites%20contain%20Web%20apps.)
 [Accessed 04 February 2022].

Rouse, M., 2020. *Hyper-Text Markup Language (HTML)*. [Online]
 Available at: <https://www.theserverside.com/definition/HTML-Hypertext-Markup-Language>
 [Accessed 08 February 2022].

Schäferhoff, N., 2017. *Sublime Text*. [Online]
 Available at: <https://www.elegantthemes.com/blog/resources/the-sublime-text-code-editor-an-in-depth-review>
 [Accessed 08 February 2022].

TutorialsPoint, 2018. *Laravel*. [Online]
 Available at: https://www.tutorialspoint.com/laravel/laravel_overview.htm
 [Accessed 07 February 2022].

TutorialsPoint, 2018. *MySQL Database*. [Online]
 Available at: <https://www.tutorialspoint.com/mysql/index.htm>
 [Accessed 08 February 2022].

W3Softech, 2019. *What is Functional Testing? Types, Tools and Examples*. [Online]
 Available at: <https://w3softech.com/blog/functional-testing/>
 [Accessed 08 February 2022].

Appendices

APPENDIX A: EMPLOYEE FEEDBACK FORM

“Google Form” to collect feedback from the client company employees, on the project progress.

The image shows a Google Form titled "Online Hotel Reservation and Centralized Management System" with the subtitle "Employee Feedback Form (Don Y.J Holdings)". The form contains several questions:

- Question 1: "Do you think the company currently requires an automated management system?" with radio button options for "Yes" and "No".
- Question 2: "What was your initial impression towards the system development?" with a 5-point Likert scale (radio buttons labeled 1 to 5).
- Question 3: "How would you rate the analysis phase of the project?" with a 5-point Likert scale (radio buttons labeled 1 to 5, with "Poor" at the start and "Excellent" at the end).
- Question 4: "How satisfied are you with the system development approaches?" with a 5-point Likert scale (radio buttons labeled 1 to 5, with "Poor" at the start and "Excellent" at the end).
- Question 5: "Do you think the system requirements are properly identified and categorized?" with radio button options for "Yes" and "No".
- Question 6: "How would you rate the user interface of the application?" with a 5-point Likert scale (radio buttons labeled 1 to 5, with "Poor" at the start and "Excellent" at the end).
- Question 7: "Additional Feedback" with a text input field labeled "Your answer".

At the bottom of the form, there are two buttons: "Submit" (in a blue box) and "Clear form" (in a light blue box).

Figure 165: Appendix A.1 - Client employee feedback form

APPENDIX B: APPLICATION USER GUIDE

Online Reservation and Centralized Management System

Don Y.J Holdings

- APPLICATION USER GUIDE -

Guest Login:

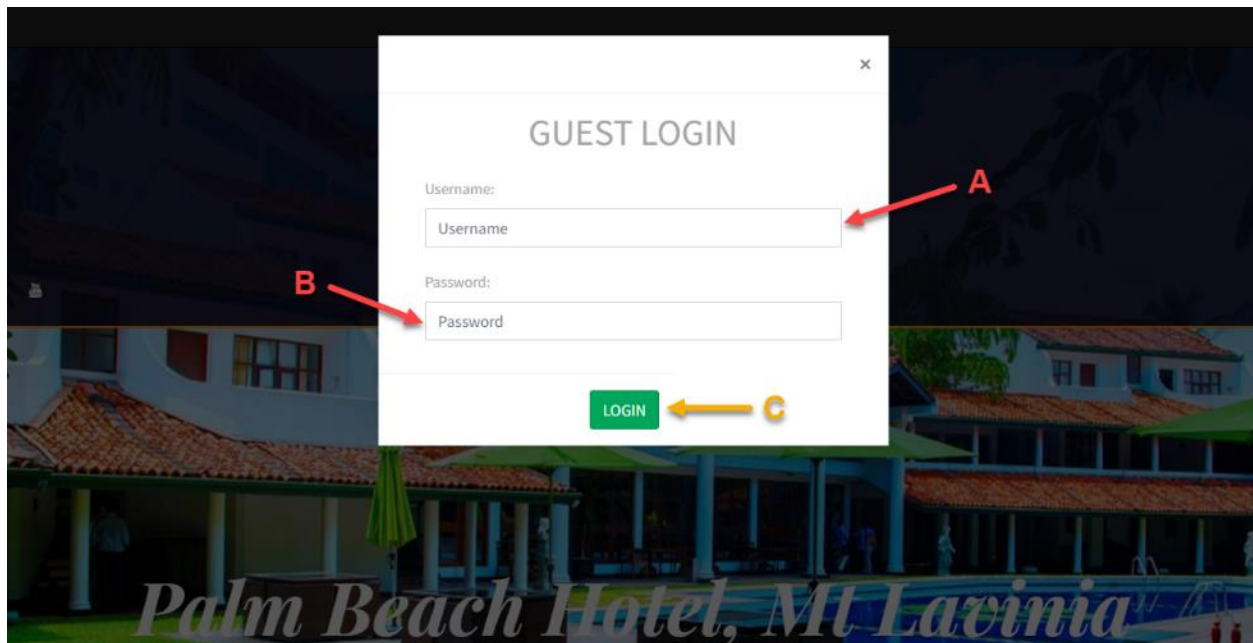
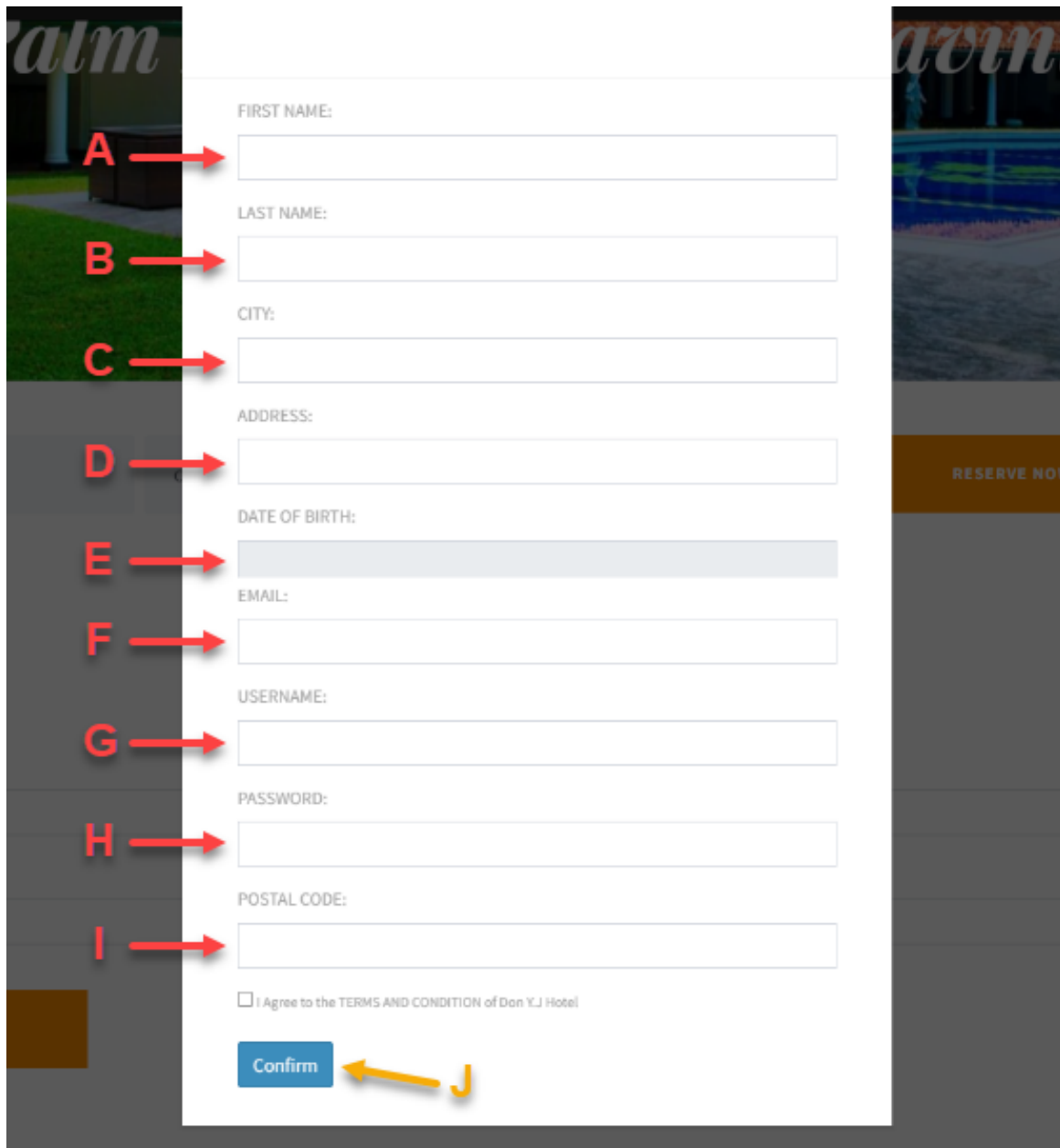


Figure 166: Appendix B.1 - Guest Login

Table 47: Appendix B.1 - Guest Login

Letter	Component	Description
A	Textbox	The guest is required to enter the specific “Username” provided during registration.
B	Textbox	The guest is required to enter the unique “Password” provided along with username during registration.
C	Button	Click the button after providing the required textbox values, which will redirect to reservation system if successful.
Login Successful:		Redirects to reservation homepage under guest profile.
Login Failed:		Error message stating “Invalid Username or Password”.
Troubleshooting:		Verify the entered username and password credentials. Verify whether values are entered in correct textbox.

Guest User Registration:



The image shows a 'Guest User Registration' form with various input fields and a 'Confirm' button. Red arrows labeled A through I point to specific fields, and a yellow arrow labeled J points to the 'Confirm' button. The form is set against a background image of a hotel lobby with the word 'Palm' visible.

A → FIRST NAME:

B → LAST NAME:

C → CITY:

D → ADDRESS:

E → DATE OF BIRTH:

F → EMAIL:

G → USERNAME:

H → PASSWORD:

I → POSTAL CODE:

☐ I Agree to the TERMS AND CONDITION of Don Y.J Hotel

J →

Figure 167: Appendix B.2 - Guest User Registration

Table 48: Appendix B.2 - Guest User Registration

Letter	Component	Description
A	Textbox	The guest is required to enter “ First Name ”.
B	Textbox	The guest is required to enter “ Last Name ”.
C	Textbox	The guest is required to enter “ City ”.
D	Textbox	The guest is required to enter “ Address ”.
E	Date Picker	The guest is required to pick “ Date of Birth ”.
F	Textbox	The guest is required to enter “ Email ”.
G	Textbox	The guest is required to enter “ Username ”.
H	Textbox	The guest is required to enter “ Password ”.
I	Textbox	The guest is required to enter “ Postal Code ”.
J	Button	Click the button after providing the required textbox values.
Action Successful:		“User registration successful” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		<p>Verify the data entered in the text fields.</p> <p>Verify whether values are entered in correct textbox.</p> <p>Refresh site and try again.</p>

Search Room Availability:



Figure 168: Appendix B.3 – Search Room Availability

Table 49: Appendix B.3 - Search Room Availability

Letter	Component	Description
A	Date Picker	The guest is required to pick the “ Check in ” date for the intended reservation availability check.
B	Date Picker	The guest is required to pick the “ Check out ” date for the intended reservation availability check.
C	Dropdown Menu	The guest is required to select the number of “ Persons ” for the intended reservation availability check.
D	Dropdown Menu	The guest is required to select the “ Accommodation ” type for the intended reservation availability check.
Action Successful:		List of rooms available for reservation would be displayed.
Action Failed:		Error message to fill the text boxes will be displayed.
Troubleshooting:		Verify if all text input fields are filled with required data. Refresh the site and try the action again.

Reservation System Admin Login:

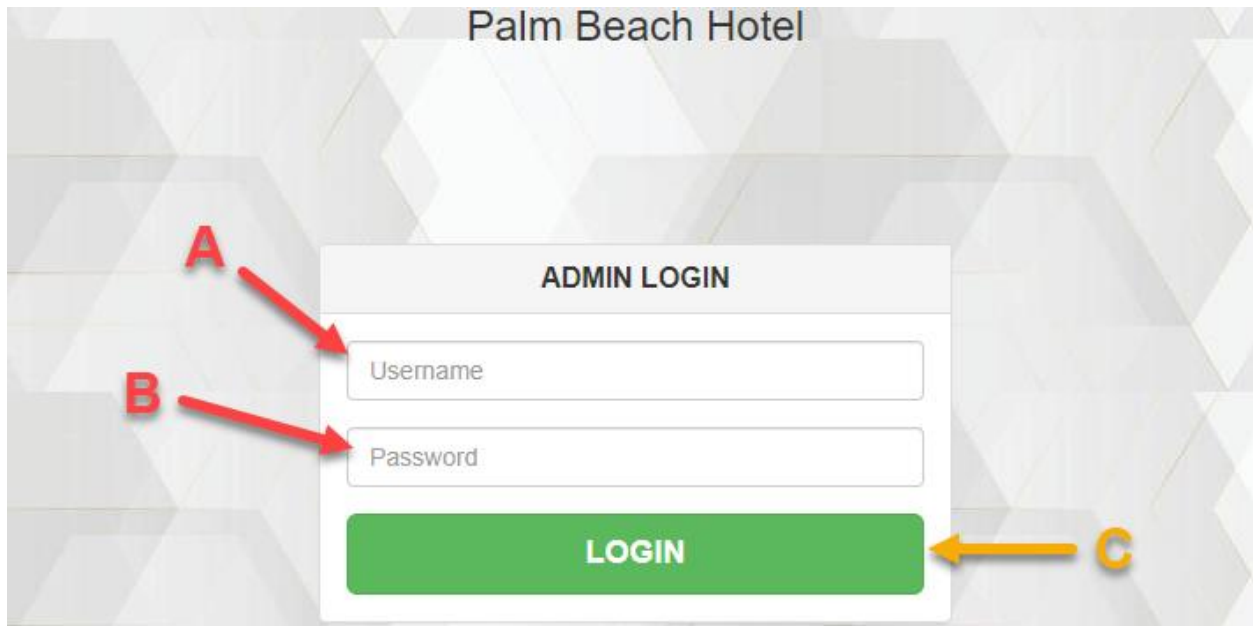


Figure 169: Appendix B.4 - Reservation System Admin Login

Table 50: Appendix B.4 - Reservation System Admin Login

Letter	Component	Description
A	Textbox	The administrator is required to enter the specific “ Username ” registered in the system.
B	Textbox	The administrator is required to enter the specific “ Password ” registered in the system.
C	Button	Click the button after providing the required textbox values, which will redirect to reservation dashboard if successful.
Login Successful:		Redirects to reservation administrator dashboard.
Login Failed:		Error message stating “Invalid Username or Password”.
Troubleshooting:		Verify the entered username and password credentials. Verify whether values are entered in correct textbox.

View and Manage Room Types:

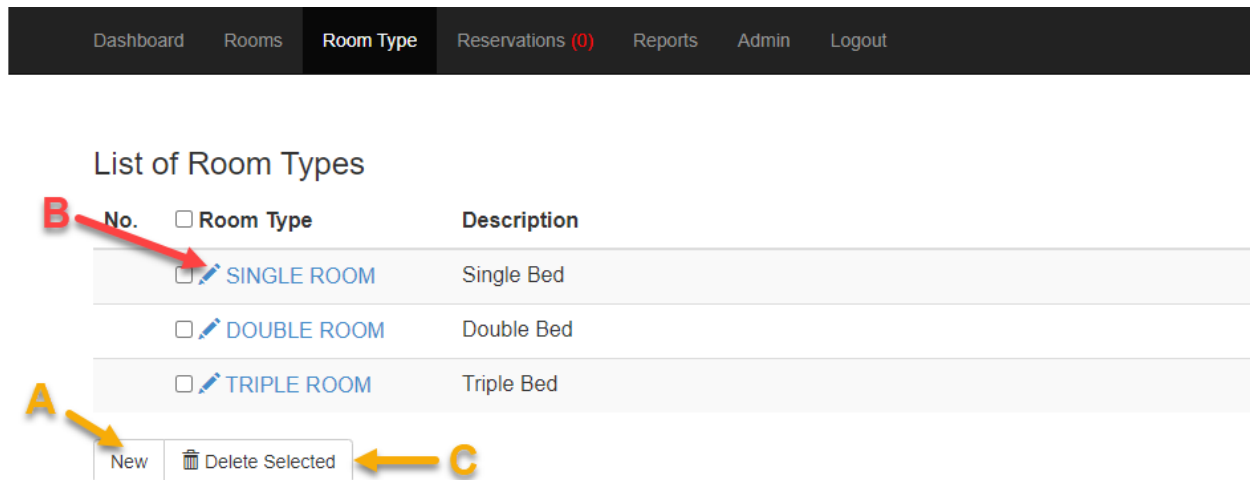


Figure 170: Appendix B.5 – View and Manage Room Types

Table 51: Appendix B.5 – View and Manage Room Types

Letter	Component	Description
A	Button	Click the button to add a new room type in the system, redirecting to add room type form.
B	Button	Click the button to edit the corresponding room type, redirecting to edit room type form.
C	Button	Tick the required records and click the button to delete the corresponding room types.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Add New Room Type:

The screenshot shows a web application interface with a dark navigation bar at the top containing links: Dashboard, Rooms, Room Type, Reservations (0), Reports, Admin, and Logout. The 'Room Type' link is highlighted. Below the navigation bar is a light gray section titled 'New Room Type'. This section contains two text input fields. The first field is labeled 'Name:' and contains the placeholder text 'Room Type'. The second field is labeled 'Description:' and contains the placeholder text 'Room Type Description'. Below these fields is a blue button labeled 'Confirm'. Three arrows are overlaid on the form: a red arrow labeled 'A' points to the 'Name' input field, a red arrow labeled 'B' points to the 'Description' input field, and a yellow arrow labeled 'C' points to the 'Confirm' button.

Figure 171: Appendix B.6 - Add New Room Type

Table 52: Appendix B.6 - Add New Room Type

Letter	Component	Description
A	Textbox	The administrator is required to enter the “ Room Type ”.
B	Textbox	The administrator is required to enter the “ Room Type Description ”.
C	Button	Click the button after providing the required textbox values.
Action Successful:		“Room type added successfully” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		<p>Verify the data entered in the text fields.</p> <p>Verify whether values are entered in correct textbox.</p> <p>Refresh site and try again.</p>

View and Manage Rooms:

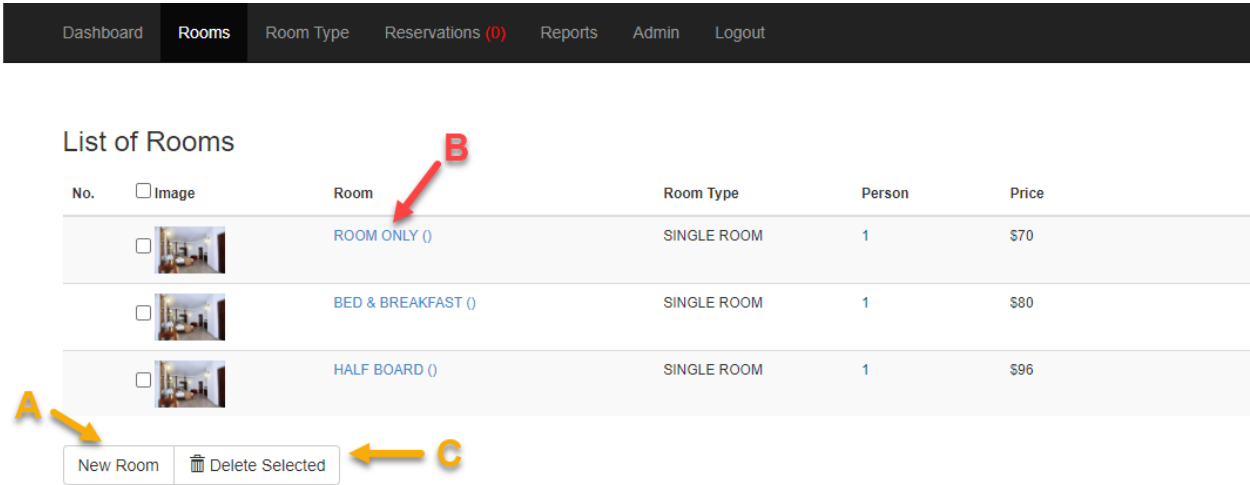


Figure 172: Appendix B.7 – View and Manage Rooms

Table 53: Appendix B.7 – View and Manage Rooms

Letter	Component	Description
A	Button	Click the button to add a new room in the system, redirecting to add room type form.
B	Button	Click the button to edit the corresponding room, redirecting to edit room type form.
C	Button	Tick the required records and click the button to delete the corresponding rooms.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Add New Room:

The screenshot shows a web application interface for adding a new room. At the top is a navigation bar with links: Dashboard, Rooms (active), Room Type, Reservations (0), Reports, Admin, and Logout. Below the navigation bar is the 'New Room' form. The form contains the following fields and components, each annotated with a letter and a red arrow:

- A**: Points to the 'Name:' label.
- B**: Points to the 'Room Type:' dropdown menu.
- C**: Points to the 'Description:' label.
- D**: Points to the 'Number of Person:' label.
- E**: Points to the 'Price (\$):' label.
- F**: Points to the 'Choose File' button in the 'Upload Image:' section.
- G**: Points to the 'Save' button at the bottom of the form.

The form fields are: Name (text input), Room Type (dropdown menu), Description (text input), Number of Person (text input), Price (\$) (text input), and Upload Image (file picker). The 'Save' button is a blue button.

Figure 173: Appendix B.8 - Add New Room

Table 54: Appendix B.8 - Add New Room

Letter	Component	Description
A	Textbox	The administrator is required to enter the “ Room Name ”.
B	Dropdown	The administrator is required to select the “ Room Type ”.
C, D, E	Textbox	The administrator is required to enter the “ Description ”, “ Number of Persons ” and “ Price ”.
F	Image Picker	The administrator is required to select a “ Image ”.
G	Button	Click the button after providing the required textbox values.
Action Successful:		“Room added successfully” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		<p>Verify the data entered in the text fields.</p> <p>Verify whether values are entered in correct textbox.</p> <p>Refresh site and try again.</p>

View and Manage Reservations:

#	Guest	Transaction Date	Confirmation Code	Total Rooms	Total Price	Status	Action
	Robert Whilliamson	2022-04-28 02:29:03	k3wtay40	1	€120	Pending	<div> <div>View</div> <div>Cancel</div> <div>Confirm</div> <div>Delete</div> <div>Edit</div> </div>
	Robert Whilliamson	2022-04-24 10:01:04	wjbn2qfa	1	€96	Checkedout	<div> <div>View</div> <div>Delete</div> <div>Edit</div> </div>

Figure 174: Appendix B.9 - View and Manage Reservations

Table 55: Appendix B.9 - View and Manage Reservations

Letter	Component	Description
A	Button	Click the button to confirm the corresponding reservation.
B	Button	Click the button to view the corresponding reservation.
C	Button	Click the button to delete the corresponding reservation.
D	Button	Click the button to edit the corresponding reservation.
E	Button	Click the button to cancel the corresponding reservation.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Booking Report Generation:

Room Booking Report Date: 04/28/2022

A →
 Status
 Checkedin ▼

B →
 Checkedin
 2022-04-28

C →
 Checkedout
 2022-04-28

Submit
D →

Guest	Room	Price	Arrival	Departure	Night(s)	Subtotal
Total Amount						
Subtotal:					\$	

E →
Print

Figure 175: Appendix B.10 - Booking Report Generation

Table 56: Appendix B.10 - Booking Report Generation

Letter	Component	Description
A	Dropdown	Select the “ Status ” of the report to be generated.
B	Date Picker	Select the “ Check-in date ” of the report to be generated.
C	Date Picker	Select the “ Check-out date ” of the report to be generated.
D	Button	Click the button to generate the report with provided filters.
E	Button	Click the button to print the generated report.
Action Successful:		Report generated and displayed with print option.
Action Failed:		Report not generated or error message
Troubleshooting:		Verify the data entered in the input fields. Refresh site and try again.

Management System Admin / Employee Login:

Don Y.J Holdings
Centralized Management

EMPLOYEE LOGIN

A → Email Address / Username

B → Password

C → LOGIN

Figure 176: Appendix B.11 - Management System Admin / Employee Login

Table 57: Appendix B.11 - Management System Admin / Employee Login

Letter	Component	Description
A	Textbox	The administrator or employee is required to enter the specific “ Username ” registered in the system.
B	Textbox	The administrator or employee is required to enter the specific “ Password ” registered in the system.
C	Button	Click the button after providing the required textbox values, which will redirect to employee dashboard if successful.
Login Successful:		Redirects to centralized management system dashboard.
Login Failed:		Error message stating “Invalid Username or Password”.
Troubleshooting:		Verify the entered username and password credentials. Verify whether values are entered in correct textbox. Contact system administrator if still unsuccessful.

View and Manage Departments:

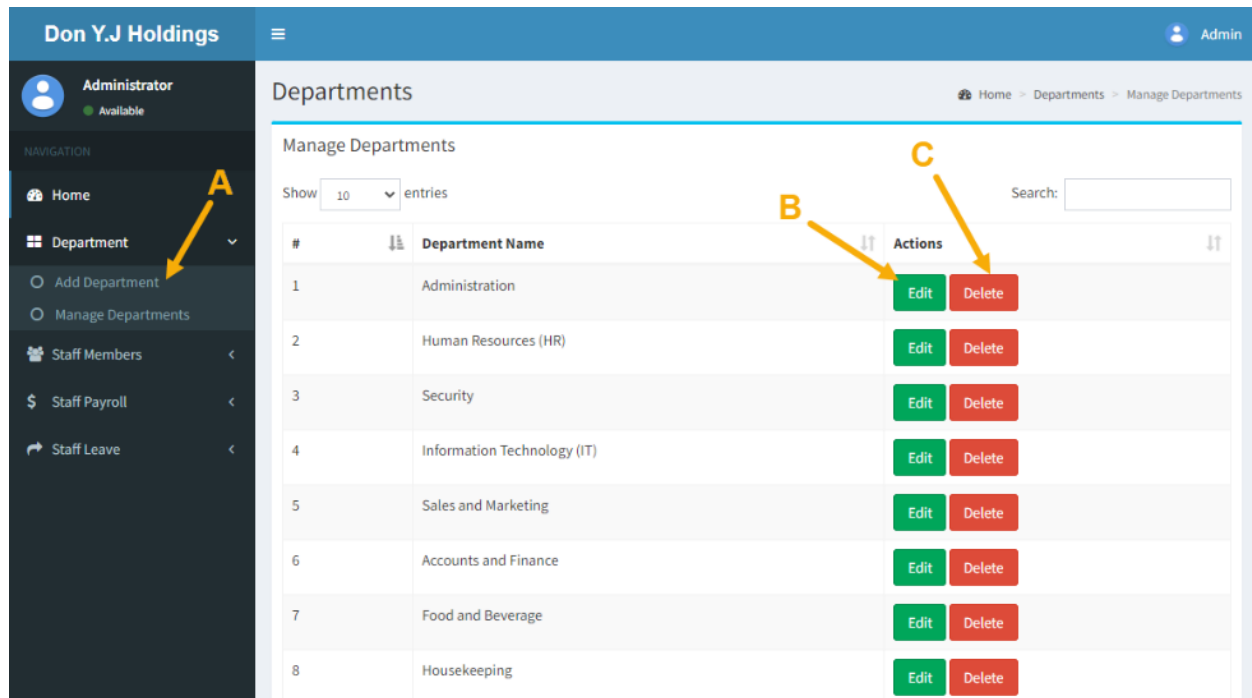


Figure 177: Appendix B.12 - View and Manage Departments

Table 58: Appendix B.12 - View and Manage Departments

Letter	Component	Description
A	Button	Click the button to add a new department in the system, redirecting to add department form.
B	Button	Click the button to edit the corresponding department, redirecting to edit department form.
C	Button	Click the button to delete the corresponding department.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Add Department:

The screenshot shows the 'Add Department' form in the DYJ Admin interface. The form has a title 'Add Department' and a label 'Department Name' above a text input field. A red arrow labeled 'A' points to the text input field. A green 'Submit' button is at the bottom right, with an orange arrow labeled 'B' pointing to it. The breadcrumb trail shows 'Home > Departments > Add Department'.

Figure 178: Appendix B.13 - Add Department

Table 59: Appendix B.13 - Add Department

Letter	Component	Description
A	Textbox	The administrator is required to enter the “ Department ”.
B	Button	Click the button after providing the required textbox values.
Action Successful:		“Department added successfully” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		Verify the data entered in the text fields. Verify whether values are entered in correct textbox. Refresh site and try again.

View and Manage Employees:

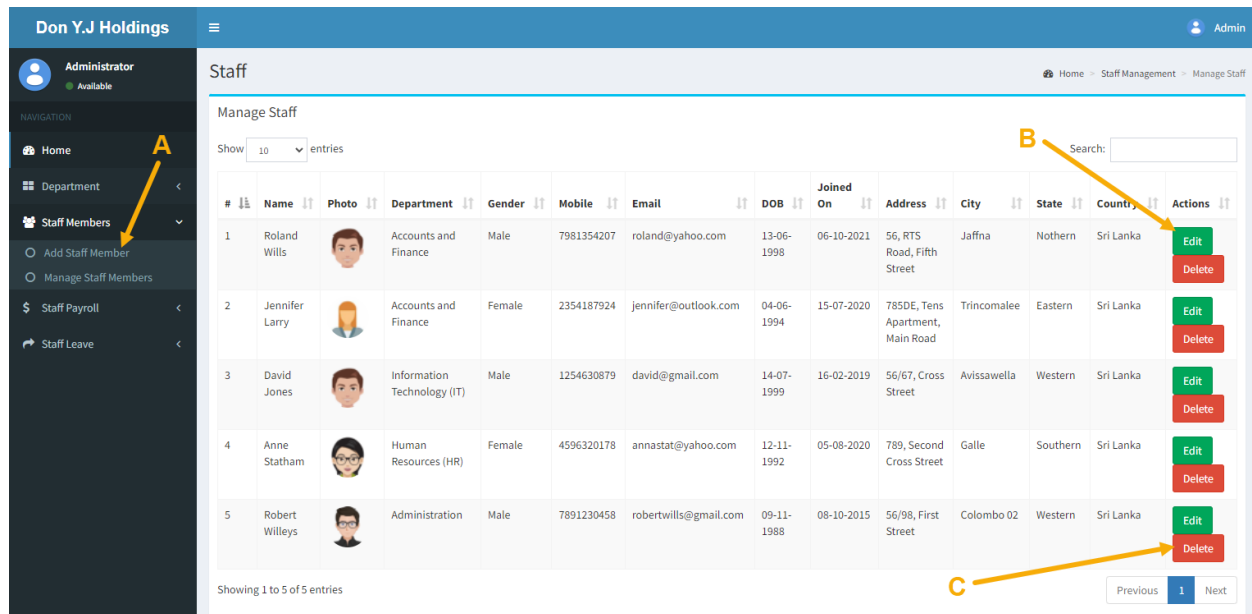


Figure 179: Appendix B.14 - View and Manage Employees

Table 60: Appendix B.14 - View and Manage Employees

Letter	Component	Description
A	Button	Click the button to add a new employee in the system, redirecting to add employee form.
B	Button	Click the button to edit the corresponding employee record, redirecting to edit employee form.
C	Button	Click the button to delete the corresponding employee record.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Add Employees:

The screenshot shows the 'Add Staff' form in the Don Y.J Holdings system. The form is titled 'Add Staff' and is part of the 'Staff Management' section. It contains various input fields for employee information, including Full Name, Department, Gender, Email, Mobile, Photo, Date of Birth, Date of Joining, City, State, Country, Password, and Address. A red box labeled 'A' highlights the main form area, and a yellow arrow labeled 'B' points to the 'Submit' button.

Figure 180: Appendix B.15 - Add Employees

Table 61: Appendix B.15 - Add Employees

Letter	Components	Description
A	Textbox, Dropdown, Date Picker and Image Picker	The administrator is required to enter the “ Full Name, Department, Gender, Email, Mobile, Photo, Date of Birth, Date of Registration, City, State, Country, Password and Address ” of the employee record to be added,
B	Button	Click the button after providing the required textbox values.
Action Successful:		“Employee record added successfully” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		Verify the data entered in the text fields. Verify whether values are entered in correct textbox. Refresh site and try again.

View and Manage Staff Payroll:

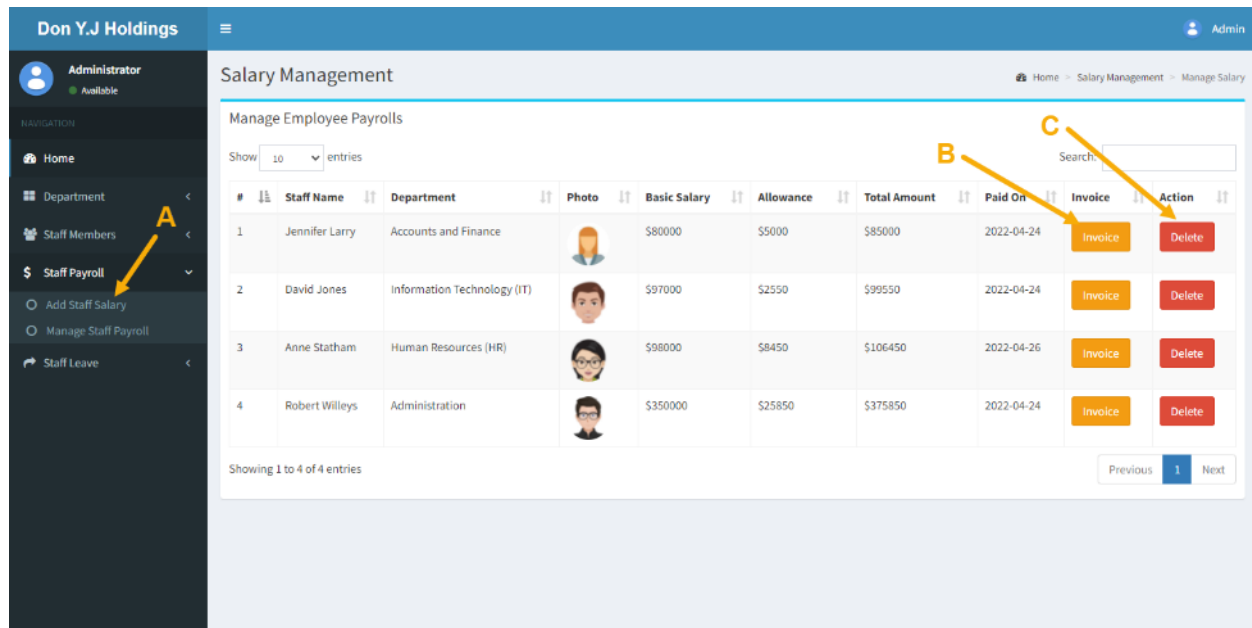


Figure 181: Appendix B.16 - View and Manage Staff Payroll

Table 62: Appendix B.16 - View and Manage Staff Payroll

Letter	Component	Description
A	Button	Click the button to add a create staff payroll in the system, redirecting to create staff payroll form.
B	Button	Click the button to generate a salary invoice on the corresponding staff salary payroll record.
C	Button	Click the button to delete the corresponding employee record.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Add Staff Salary:

Figure 182: Appendix B.17 - Add Staff Salary

Table 63: Appendix B.17 - Add Staff Salary

Letter	Component	Description
A	Dropdown	The administrator is required to select the “ Department ”.
B	Textbox	The administrator is required to enter the “ Basic Salary ”.
C	Textbox	The administrator is required to enter the “ Allowance ”.
D	Textbox	The administrator is required to verify the “ Total ”.
E	Button	Click the button after providing the required textbox values.
Action Successful:		“Staff salary added successfully” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		<p>Verify the data entered in the text fields.</p> <p>Verify whether values are entered in correct textbox.</p> <p>Refresh site and try again.</p>

View and Manage Staff Leave Requests:

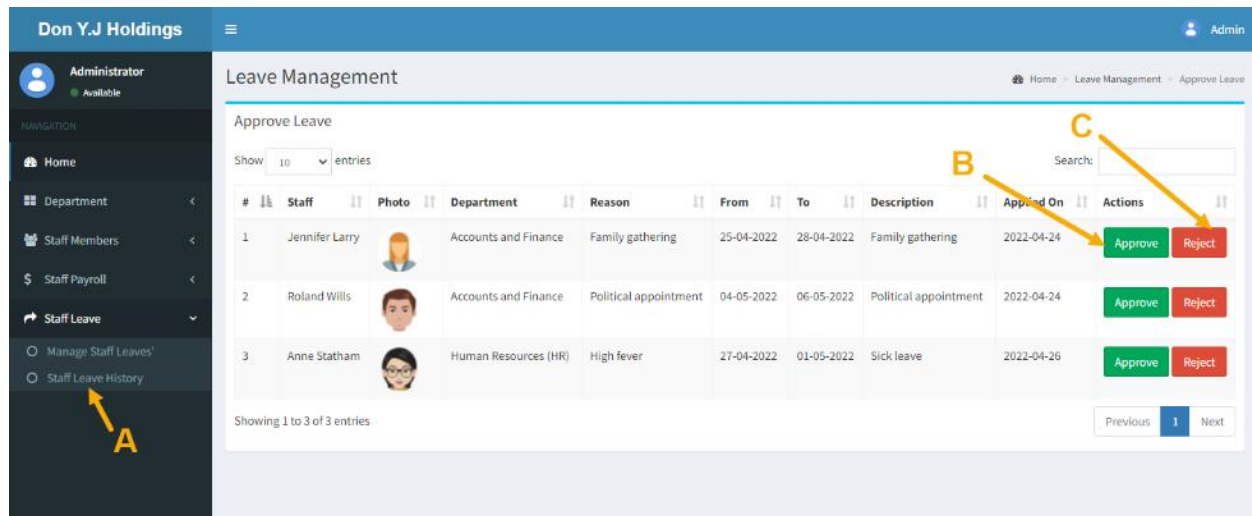


Figure 183: Appendix B.18 - View and Manage Staff Leave Requests

Table 64: Appendix B.18 - View and Manage Staff Leave Requests

Letter	Component	Description
A	Button	Click the button to view a history of staff leave requests.
B	Button	Click the button to “ Approve ” the corresponding leave request.
C	Button	Click the button to “ Reject ” the corresponding leave request.
Action Successful:		Relevant component functionality would be implemented.
Action Failed:		Error messages or system malfunction.
Troubleshooting:		Verify whether proper actions are being selected. Refresh the site and try the action again.

Create New Staff Leave Request:

Figure 184: Appendix B.19 - Create New Staff Leave Request

Table 65: Appendix B.19 - Create New Staff Leave Request

Letter	Component	Description
A	Button	Click the button to view the specific staff leave request history.
B	Textbox	The employee is required to enter the “ Reason ”.
C	Date picker	The employee is required to select the “ Leave Start ” date.
D	Date picker	The employee is required to select the “ Leave End ” date.
E	Textbox	The employee is required to enter the “ Description ”.
F	Button	Click the button after providing the required textbox values.
Action Successful:		“Staff leave request added successfully” message.
Action Failed:		Error message stating “All fields required” or no response.
Troubleshooting:		<p>Verify the data entered in the text fields.</p> <p>Verify whether values are entered in correct textbox.</p> <p>Refresh site and try again.</p>