EVENT MANAGEMENT AND RESOURCING PLATFORM (EVENZA)

By Madushanka H.W.C

(IM/2016/046)

A report submitted in partial fulfillment of the requirements for the degree of Bachelor of Science (Honours) in Management and Information Technology (B.Sc. MIT)

Name of the Supervisor: Dr. Dilani Wickramarachchi

Department of Industrial Management

University of Kelaniya

Sri Lanka

2020

DECLARATION

I hereby certify that this project and the all the artifacts associated with it is my own work and it has not been submitted before nor is currently being submitted for any other degree programme.

Full name of the student: Hewa Witharana Chiran Madushanka

Student No: IM/2016/046

Signature of the student: …….. Date: 23-08-2020

Name of the supervisor(s): Dr. Dilani Wickramarachchi

Signature of the supervisor: …….. Date: 23-08-2020

ACKNOWLEDGEMENT

I express my sincere gratitude to my Project Supervisor, Dr. Dilani Wickramarachchi for the continuous support and for being the pillar of strength in succeeding the completion of the event management and resourcing platform (Evenza).

I thank my Project Coordinator Mr.Janaka Senanayake for the immense helping hand given throughout. I would also like to express my gratitude to Mr. Pasindu De Silva (Biz Lead and coordinator – Ideamart for Dialog Axiata PLC) for providing me with the required information and for the support given.

I convey my heartiest gratitude to every individual who helped me in making this project a success.

ABSTRACT

Event management is the application to manage and development of festivals, events, celebrations and conferences. Proposed work involves study of identifying the target of budget, cost, and analysis. Pre event analysis and ensuring a return on investment have become significant drivers for the event industry. Lack of a platform for event related service providers to promote their business to the exact customer base has become also a significant issue with the industry.

The proposed system is an online event management and resourcing platform that serves the functionality of an event manager as well as providing access to event related resources by connecting relevant service providers to the customers .The core uniqueness is that proposed system provides a solution for planning and management of an event as well as finding necessary resources (services and products) within the same platform. The project also provides most of the basic functionality required for an event. It consists of the current background of the problems we face while planning for upcoming events and how planners can tackle these matters through this proposed system. It also entails building a fully functional system that would help achieve a well-organized, timely and within the budget event.

This web and mobile based platform can be integrated with venue services as hotels, clubs, auditoriums and other event related service providers as coordinators, choreographers, caters, photographers, renters, dress makers etc. as resources. The user gets all the resources at a single place instead of wondering round for these. This system is effective and saves time and cost for the user.

TABLE OF CONTENTS

DECLARATION i
ACKNOWLEDGEMENT ii
ABSTRACT iii
TABLE OF CONTENTS iv
LIST OF FIGURES x
LIST OF TABLES xiv
ABBREVIATIONS xv
1 INTRODUCTION 1
1.1 Event Management and Resourcing in Sri Lanka 1
1.2 Introduction to “Evenza” online Event Management and Resourcing Platform 3
1.3 Problem Definition 4
1.4 Current Business Process 5
1.5 Aims and Objectives 6
1.6 Scope of the project 7
1.7 Tools and Project Feasibility 8
1.7.1 Technical Feasibility 8
1.7.2 Economical Feasibility 8
1.7.3 Operational Feasibility 8
1.7.4 Organizational Feasibility 8
1.8 Organization of the Report 9
1.9 Summary 9
2 SYSTEM ANALYSIS 10
2.1 Requirement Analysis 10
2.2 Use Case Diagram for the Current System 11
2.3 Activity Diagrams for the Current System 14
2.3.1 Activity Diagram for creating an event plan 14
2.3.2 Activity Diagram for finding required service providers 15
2.3.3 Activity Diagram for finding required product suppliers 16
2.3.4 Activity Diagram for booking a service 17
2.3.5 Activity Diagram for create a visit appointment for a venue service 18
2.3.6 Activity Diagram for purchase products for the event 19
2.3.7 Activity Diagram for inviting and managing event participants 20

2.4 System Requirement Analysis....21
2.4.1 Functional Requirements....21
2.4.2 Non-functional Requirements....22
2.4.3 Functional Requirement explanation....23
2.5 Business System Options....27
2.5.1 Business system option identification....27
2.5.2 BSO1- Web based event management and resourcing system with Progressive mobile web application (PWA)....28
2.5.3 BS02 – Web based system with android and iOS based native mobile applications....30
2.5.4 BSO3 - Web based application which utilize an existing event planning platform using its API (Ex: - eventray, openevent)....32
2.6 Evaluation of BSOs....34
2.6.1 Functional requirements Vs. BSOs....34
2.6.2 Non-functional requirements Vs. BSOs....36
2.6.3 Selected BSO with a sound justification....36
2.7 Summary....37
3 SYSTEM DESIGN....38
3.1 Use Case Diagram for the Proposed System....38
3.1.1 UC92 Use Case for update user profile Figure 3.2 illustrates the use case diagram for update user profile. The activities are done by both customer and Merchant....40
3.1.2 UC110 Use Case for create an event plan....40
3.1.3 UC120 Use Case for search and sort products or services....41
3.1.4 UC140 Use Case for add products and services....42
3.1.5 UC160 Use Case for place a visit appointment for a service....43
3.1.6 UC180 Use Case for send invitations to the ‘closed type’ event invitees....44
3.1.7 UC200 Use Case for generate reports....45
3.1.8 UC210 Use Case for collect commission and subscription fee by System Administrator....45
3.1.9 UC220 Use Case for rate and review services/ products....46
3.2 Use Case Descriptions for the Proposed System....47
3.2.1 UC92 Use case description for update user profile....47
3.2.2 UC110 Use case description for create an event plan....48
3.2.3 UC120 Use case description for search and sort services or products....49
3.2.4 UC130 Use case description for create a business profile....50

3.2.5 UC140 Use case description for add services or products to the business profile 51
3.2.6 UC150 Use case description for place a booking for a service 52
3.2.8 UC170 Use case description for place an order for a required product 54
3.2.9 UC180 Use case description for send invitations to the closed type event invitees 56
3.2.10 UC190 Use case description for get random participants registered to the open type events 57
3.2.11 UC200 Use case description for generate reports 58
3.2.12 UC210 Use case description for collect commission and subscription fee by system administrator 59
3.2.13 UC220 Use case description for rate and review services and products 60
3.3 Activity Diagrams for the Proposed System 61
3.3.1 Activity diagram for sign up to the system 61
3.3.2 Activity diagram for log in to the system 62
3.3.3 Activity diagram for update Merchant's user profile 63
3.3.4 Activity diagram for generate event plan for available event category 64
3.3.5 Activity diagram for and generate event plan for new event category 65
3.3.6 Activity diagram for creating business profile 66
3.3.7 Activity diagram for add and modify products and services 67
3.3.8 Activity diagram for search and filter products and services 68
3.3.9 Activity diagram for place a booking for a service 69
3.3.10 Activity diagram for place a visit appointment for a venue service 70
3.3.11 Activity diagram for place an order for a required product 71
3.3.12 Activity diagram for send invitations to invitees (closed type events) 71
3.3.13 Activity diagram for get participants registered (open type events) 73
3.3.14 Activity diagram for generating reports 74
3.3.15 Activity diagram for collect commission by admin 75
3.3.16 Activity diagram for manage users by admin 76
3.3.17 Activity diagram for chat with chatbot 77
3.4 Sequence Diagrams for the Proposed System 78
3.4.1 Sequence diagram for sign up to the system 78
3.4.2 Sequence diagram for log in to the system 79
3.4.4 Sequence diagram for create business profile 79
3.4.5 Sequence diagram for create event plan 81
3.4.6 Sequence diagram for place a booking for a service 81
3.4.7 Sequence diagram for place a visit appointment for venue service 82

3.4.8 Sequence diagram for order a product 83
3.5 Class Diagram for Proposed System 84
3.6 Database Design 86
3.6.1 Entity – Relationship model 86
3.6.2 Normalized database diagram 87
3.7 Proposed GUI Design 88
3.7.1 Welcome page GUI 89
3.7.3 System Sign up GUIs (Merchant's sign up) 90
3.7.5 GUI of events overview 92
3.7.6 GUIs for creating an event plan 92
3.7.7 GUI of Merchant's business profile 94
3.7.8 GUI of adding services to the business profile 94
3.7.9 GUI of adding products to the business profile by a product supplier 95
3.7.10 GUI of Service overview and booking 95
3.7.11 GUI of payment page 96
3.7.12 GUI of managing closed type event invitees 96
3.7.13 GUI of creating registration form for open type event participants 97
3.7.14 GUI of generating reports 97
3.7.15 GUI of Bookings overview of a customer 98
3.7.16 GUIs of mobile interface - Getting started page 98
3.7.17 GUIs of mobile interface - Service list 98
3.7.18 GUIs of mobile interface - sidebar navigation 99
3.7.19 GUIs of mobile interface - chat with chatbot 99
3.7.20 GUIs of mobile interface - Receive Notifications 100
3.8 System Reports 101
3.9 Summery 101
4 SYSTEM DEVELOPMENT 102
4.1 Programming languages 102
4.1.1 Hypertext Markup Language (HTML) 102
4.1.2 Cascade Style Sheets (CSS) 103
4.1.3 Javascript 103
4.1.4 Typescript 104
4.1.5 AJAX 105
4.2 Development Tools and Technologies 105
4.2.1 Angular 105
4.2.2 Express JS 106

4.2.3 Node JS 106
4.2.4 MongoDB.....106
4.2.5 Mongoose 107
4.2.6 Visual Studio Code IDE 107
4.3 Third party Components/ Libraries 107
4.3.1 JWT (JSON Web Token) 107
4.3.2 Socket.io 108
4.3.3 Multer.....108
4.3.4 Nodemailer.....108
4.3.5 Angular Material: UI elements 109
4.3.6 MDB UI component library 109
4.3.7 Bodyparser 109
4.2.8 Ngx-qrcode 109
4.3.9 rxjs 110
4.3.10 Ngx-cron 110
4.3.11 nodemon 110
4.4 API services and other services.....110
4.4.1 Google Maps API 110
4.4.1 Paypal Sandbox 111
4.4.2 Mongodb Atlas 111
4.5 Design Principles and Patterns 111
4.5.1 Singleton Pattern 111
4.5.2 Observer Design Pattern 112
4.6 Summary.....112
5 SYSTEM TESTING 113
5.1 Test Plan and Test strategy.....113
5.1.1 Testing Approach 113
5.2 Sample Test Cases 115
5.3 Test Report 125
5.3.1 Test Environment 125
5.4 Summary.....126
6 SYSTEM IMPLEMENTATION.....127
6.1 System Requirements.....127
6.1.1. Hardware Requirements 127
6.1.2 Software Requirements 127
6.2 Installation guide 128

6.3 User Access Levels 129
6.3.1 Admin 129
6.3.2 Merchants (Service Providers/ Sellers) 129
6.3.3 Customer 129
6.4 User Manual 130
6.4.1 Customer (Event Planner) User Manual 130
6.4.2 Merchant (Service Providers and Sellers) User Manual.....143
6.4.3 System Administrator (Owner) User Manual.....154
7 EVALUATION AND CONCLUSION 159
7.1 Degree of Objectives Met.....159
7.2 Cross Checking the Objectives.....160
7.2.1 Functional Requirements 160
7.3 Usability, Accessibility, Reliability and Friendliness 161
7.4 Limitations and Drawbacks.....162
7.5 Future Modification, Improvements and Extensions.....163
7.6 Summary 163
REFERENCES 164

LIST OF FIGURES

Figure 1.1 Scope of the Project 7
Figure 2.1 UC10 overall use case diagram for current business process 12
Figure 2.2 Activity Diagram for creating an event plan 14
Figure 2.3 Activity Diagram for finding required service providers 15
Figure 2.4 Activity Diagram for finding required product suppliers 16
Figure 2.5 Activity Diagram for create a booking for a service 17
Figure 2.6 Activity Diagram for create a visit appointment for a venue service 18
Figure 2.7 Activity Diagram for purchase required products for the event 19
Figure 2.8 Activity Diagram for inviting and managing event participants 20
Figure 3.1 UC20 Overall use case for the proposed system 39
Figure 3.2 UC92 Use Case for update user profile 40
Figure 3.3 UC110 Use Case for create event plan 40
Figure 3.4 UC120 Use Case for search and sort products or services 41
Figure 3.5 UC130 Use Case for create business profile 41
Figure 3.6 UC140 Use Case for add products and services 42
Figure 3.7 UC150 Use Case for place a booking for a service 42
Figure 3.8 UC160 Use Case for place a visit appointment for a service 43
Figure 3.9 UC170 Use Case for place an order for a required product 43
Figure 3.10 UC180 Use Case for send invitations to the closed type event invitees .. 44
Figure 3.11 UC190 Use Case for get random participants registered to the open type events 44
Figure 3.12 UC200 UC200 Use Case Diagram for generate reports 45
Figure 3.13 UC210 Use Case for collect commission and subscription fee 45
Figure 3.14 UC170 Use Case for rate and review services/ products 46
Figure 3.15 Activity diagram for sign up to the system 61
Figure 3.16 Activity diagram for log in to the system 62
Figure 3.17 Activity diagram for update Merchant's user profile 63
Figure 3.18 Activity diagram for generate event plan for available event category 64
Figure 3.19 Activity diagram for and generate event plan for new event category 65
Figure 3.20 Activity diagram for creating business profile 66
Figure 3.21 Activity diagram for add and modify products and services 67
Figure 3.22 Activity diagram for search and filter products and services 68

Figure 3.23 Activity diagram for place a booking for a service 69
Figure 3.24 Activity diagram for place a visit appointment for a venue service..... 70
Figure 3.25 Activity diagram for place an order for a required product..... 71
Figure 3.26 Activity diagram for send invitations to invitees (closed type events) 72
Figure 3.27 Activity diagram for get participants registered (open type events) 73
Figure 3.28 Activity diagram for generating reports 74
Figure 3.29 Activity diagram for collect commission by admin 75
Figure 3.30 Activity diagram for manage users by admin..... 76
Figure 3.31 Activity diagram for chat with chatbot..... 77
Figure 3.32 Sequence diagram for sign up to the system 78
Figure 3.33 Sequence diagram for log in to the system..... 79
Figure 3.34 Sequence diagram for create business profile 80
Figure 3.35 Sequence diagram for create event plan 81
Figure 3.36 Sequence diagram for place a booking for a service 82
Figure 3.37 Sequence diagram for place a visit appointment for venue service 83
Figure 3.38 Sequence diagram for order a product..... 84
Figure 3.39 Class Diagram for the proposed system 85
Figure 3.40 Entity Relationship diagram for the proposed system..... 86
Figure 3.41 Normalized database design for the proposed system..... 87
Figure 3.42 Welcome page GUI 89
Figure 3.43 System Sign in GUI..... 89
Figure 3.44 System Sign up GUIs (Merchant’s sign up) 01..... 90
Figure 3.45 System Sign up GUIs (Merchant’s sign up) 02..... 90
Figure 3.46 System Sign up GUIs (Merchant’s sign up) 03..... 91
Figure 3.47 GUI of user dashboard..... 91
Figure 3.48 GUI of events overview..... 92
Figure 3.49 GUIs for creating an event plan 01..... 92
Figure 3.50 GUIs for creating an event plan 02..... 93
Figure 3.51 GUIs for creating an event plan 03..... 93
Figure 3.52 GUI of Merchant’s business profile 94
Figure 3.53 GUI of adding services to the business profile..... 94
Figure 3.54 GUI of adding products to the business profile by a product supplier..... 95
Figure 3.55 GUI of Service overview and booking..... 95
Figure 3.56 GUI of payment page 96

Figure 3.57 GUI of managing closed type event invitees 96
Figure 3.58 GUI of creating registration form for open type event participants 97
Figure 3.59 GUI of generating reports 97
Figure 3.60 GUI of Bookings overview of a customer 98
Figure 3.61 GUIs of mobile interface - Getting started page 98
Figure 3.62 GUIs of mobile interface - Service list 99
Figure 3.63 GUIs of mobile interface - sidebar navigation 99
Figure 3.64 GUIs of mobile interface - chat with chatbot 100
Figure 3.65 GUIs of mobile interface - Receive Notifications 100
Figure 6.1 Evenza homepage 132
Figure 6.2 Sign up type select screen 132
Figure 6.3 User sign Up Details 133
Figure 6.4 User Login Screen 100
Figure 6.5 Evenza user Dashboard 134
Figure 6.6 Select Event Category 134
Figure 6.7 Create an Event - step 1 135
Figure 6.8 Create an Event - Step 2 135
Figure 6.9 Create an Event - Select Service and Product Categories required 136
Figure 6.10 Create an event - Summary view before creation 136
Figure 6.11 event Details View 100
Figure 6.12 Event Plan: Task List 100
Figure 6.13 Create New task dialog Box 138
Figure 6.14 Event Plan : Service and Product Bookings, purchases & allocations with budget 139
Figure 6.15 Event Schedule 140
Figure 6.16 Participant Management: Invitation 100
Figure 6.17 Adding Guests and the guest List View 141
Figure 6.18 Event Publish Dialog 141
Figure 6.19 Service Search 142
Figure 6.20 Product Search 142
Figure 6.20 Product Search 143
Figure 6.22 Event Report Dashboard 100
Figure 6.23 Cancel Event Dialog 100
Figure 6.26 Sign up type select screen 145

Figure 6.27 Sign up: Provider user details 145
Figure 6.28 User Login Screen 146
Figure 6.29 Service Provider Dashboard 147
Figure 6.30 ID Verification required before creating a business profile 147
Figure 6.31 Submit NIC Copies for verification 148
Figure 6.32 Create the Business Profile 148
Figure 6.33 Profile view after creation 149
Figure 6.34 Set Business Opened Days 149
Figure 6.35 Setting Business Location 150
Figure 6.36 Add new Service 100
Figure 6.37 Adding Promotion to a Product or service 100
Figure 6.38 View List of Bookings 151
Figure 6.39 Cancel an order 152
Figure 6.40 Booking Calendar 153
Figure 6.41 Payment section 100
Figure 6.42 Pay subscription Fee 154
Figure 6.43 Handle Delivery Services and Product Inventory 154
Figure 6.44 Generating Business Reports 155
Figure 6.45 Admin can also sign in as usual with their credentials 156
Figure 6.46 View Overall Transaction Visualizations 157
Figure 6.47 View and Manage Subscription Payments 100
Figure 6.48 view and Manage Service Providers/ Sellers 100
Figure 6.49 Create and manage Service/ Product and Event Categories 158
Figure 6.50 Create Backups and restore 159
Figure 6.51 Verify ID and Business Verification requests 159
Figure 6.52 View and manage overall dashboards and business reports 160

LIST OF TABLES

Table 2.1 Functional Requirements 21
Table 2.2 Non - functional requirements 22
Table 2.3 Planned cost - BSO1 29
Table 2.4 Planned cost - BSO2 31
Table 2.5 planned cost - BSO3 33
Table 2.6 Functional requirements vs. BSOs 34
Table 2.7 Non-functional requirements vs. BSOs 36
Table 2.8 BSO Justification 37
Table 3.1 UC92 Use case description for update user profile 47
Table 3.2 UC110 Use case description for create an event plan 48
Table 3.3 UC120 Use case description for search and sort services or products 49
Table 3.4 UC130 Use case description for create a business profile 50
Table 3.5 UC140 Use case description for add services or products to the business profile 51
Table 3.6 UC150 Use case description for place a booking for a service 52
Table 3.7 UC160 Use case description for place a visit appointment for a venue service 53
Table 3.8 UC170 Use case description for place an order for a required product 54
Table 3.9 UC180 Use case description for send invitations to closed type event invitees 56
Table 3.10 UC190 Use case description for get participants registered to events 57
Table 3.11 UC200 Use case description for generate reports 58
Table 3.12 UC210 Use case description for collect subscription fee by system admin 59
Table 3.13 UC220 Use case description for rate and review services and products ... 60
Table 5.1 Test Case: User Registration 117
Table 5.2 Test Case: User Login 118
Table 5.3 Test Case: Creating a Business Profile 119
Table 5.4 Test Case: Adding a new Service 120
Table 5.5 Test Case: Add a new Product 121
Table 5.6 Test Case: Create a new booking 122

Table 5.7 Test Case: Create a new Visit Appointment 123
Table 5.8 Test Case: Create a new Product Order 124
Table 5.9 Creating a new event 125
Table 7.1 Cross Checking objectives met with Functional Requirements 162

ABBREVIATIONS

- GUI – Graphical User Interface
- SME – Small and Medium Scale Businesses
- Evenza – Online event management and resourcing platform.
- SRS – System Requirement Specification
- BSO – Business System Options

CHAPTER 1

1 INTRODUCTION

This chapter outlines the introduction of the business process of event management and resourcing, nature of the business, the current business process and issues. Furthermore, it analyses the objectives and aims of the proposed system, scope and boundaries and the organization of the dissertation.

1.1 Event Management and Resourcing in Sri Lanka

Event organization and management has become one of the largest contests in Sri Lanka. In a social aspect every and each of the religious gatherings, social gathering from a scale from islandwide to a single-family unit, and in a business aspect almost all of business entities are planning, organizing and managing several events. So, the event management industry has become a rapidly growing industry with increasing demand which is lack of enough technical support for its exponential development.

There are a several event planning and management organizations are currently operating in Sri Lanka. Some of them a large-scale organization where others are SMEs and startups. According to the World Bank statistics 42.845% of Total-employed in Sri Lanka were self-employed. Sri Lanka has more than 500,000 SMEs, each employing three to five persons on average. A considerable number of these SMEs are into event management industry. There are several service individuals and firms which do their business in event related resourcing. Starting form event planners and coordinators, catering services, cab services, photographers and videographers, renting services, handicraft designers, graphic designers, dress makers, decorators and dressing saloons etc. Event related product manufacturing and reselling is identified as another considerable industry related to the event management which consist of specific product suppliers as souvenirs, photo framing, food products etc.

Event some of event computer base event management platforms are available internationally, event management services in Sri Lanka are mostly operated

manually or semi-autonomous. People tend to plan events on their own finding required service suppliers and product suppliers on their own. In most cases managing large scale event, they hire event planners and event management firms or individuals. They also do their management operations manually with the aid of some automated tools like planners and schedulers. One of the major event which is planned through those event planner firms are wedding ceremonies, conferences and corporate meetups. Within the market research, I've identified several event planning and management firms as listed below which are still do their operations manually with minimum support of technology.

- Vibes Events
- Beyond boundaries Global (Pvt) Ltd.
- Colombo Conferences
- 3N Events
- CEC Events
- Imagine Entertainments
- AI Events
- Showtown entertainment
- Magical Moments (Pvt) Ltd

I also identified some major event participant management platforms for participant registration and management. Several platforms are already operates online within the International as well as Sri Lankan contest.

- Open Events
- Eventray by APACHE
- Block Events (By Dialog Axiata PLC)
- Eventbrite
- Meetups.com
- RSVP.lk

1.2 Introduction to “Evenza” online Event Management and Resourcing Platform

Event refers to an occurrence happening at a determinable time and place. It is also something significant or notable that usually happens a planned occasion or activity. Event planning is, therefore, the process of managing an occurrence or happening such as a party, tradeshow, ceremony, team building activity, or convention. Event planning entails budgeting, setting up timelines, selecting and booking the event sites, acquiring permits, planning for food, coordinating transport, coming up with a theme among others.

Event planning consists of coordinating every detail of meetings and convention, from the speakers and venue to organization of printed materials and audio-visual equipment. It starts with determining the objective that the sponsoring organization wants to achieve. Organizers choose a public address, entertainment, content and arrange the program to present the organization's information in the most efficient way. The organizers are responsible for selecting, prospective attendees and how to get them to the meeting.

Therefore, Event Planning is an uphill and demanding task. Planners must multi-task, face numerous deadlines and orchestrate the activities of many diverse groups of people. Organizers may need to travel extensively to attend meetings and to visit prospective sites and meet suppliers. Working durations can be irregularly wrong and work for more than 40 hours per week is common especially when the event is approaching.

Thinking of the hustles and baffles involved when one must go through when planning for an upcoming event, I have come up with the “Evenza” is a online event planning, management and resourcing platform which addresses all the services required regard to event management including its participant management which will be introduced to the Sri Lankan event industry for the first time. The platform mainly connects the event planners or the customers with the event resources as service providers and product suppliers directly through one platform. The uniqueness of the platform is that it is capable of providing an auto generated and fully customizable event plan according to each event and the user is allowed to tackle the plan set

reminders and most importantly find and book required services and order necessary products at the very first stage throughout the same system with no hesitate.

The platform basically handles the job of an event planner so the end user (customer) can plan their events without the support of an event planner or an event planning and management organization can use this as a tool for their growth.

A successful event is the aim of every event planner. An extraordinary event is always admired and talked. Therefore, if one wants his or her event to get remembered one must do it plan it and manage it in a different manner. The system must go through a series of steps as indicated earlier. However, some challenges such as time and budgetary constraints get experienced.

"Evenza" will cut off time wastage, energy, and money that might be used unnecessarily to move from one point to another looking for suitable items and services. All that is required is for the event planner or the customer to log in then select an event of concern to him or her such as a wedding or a birthday etc. The generated event plan gets directed to pages of shops or suppliers offering the required items or services. He or she will then select what should go into his or her event. All the provided service and product suggestions are allocated according to the event budgets and other parameters provided by the event planner or the customer.

1.3 Problem Definition

Event planning is an uphill task to many people. As much as one would wish to have a perfect occasion, several factors that are likely to hamper the process of event planning. One of these factors include; might be Shrinky budgets. Anyone dreaming of a successful event should allocate adequate funds to have it sail through, but with the tough economic times, it's become a challenge to utilize the limited resources. Time constraints is also another major factor. Planning for an occasion needs a lot of time. Time wastage is because of consultations with different suppliers who offer various goods and services. Poor planning skills also will lead to a substandard event. This kind of poor planning is mainly brought about by the lack of exposure. When the planner is not well vast with event organizing, he or she tends to make numerous mistakes that may lead to a poorly planned event.

Even if the manual planning and resourcing process goes well, it is a high cost process as contacting and booking event resources are still operated manually. Transport cost as well as the consumption of time has become a huge drawback of the current process.

The currently available event planning and resourcing process is fully or partially manual. Support of computerized systems are only used in event task planners and scheduler software which are used by event planners. When it comes to resourcing there is no platform to provide event resourcing features as well as the participant management feature within the same platform.

Lack of a proper marketplace for event related service providers and product suppliers which is focused on their customers is another major problem found. Once such marketplace is provided, they will be capable of direct target marketing as well as increasing their sales and revenue.

1.4 Current Business Process

Current business process of the event planning and resourcing is operated manually and semi-autonomous with the aid of some currently available tools. An event planner or the customer by himself first identify the event to be planned and its type and gather a set of requirements. Then they need to create a event plan manually on a paper or a event scheduler or task planner tool with identified requirements. Then required services and products should be identified and completion date for each task should be identified and set manually. Once the task date is set reminders can be added with the support of available tools.

Then the planner or the customer need to identify relevant product and service providers. The methods they use, can be identified as the available databases in the event planning organizations, recommendations, social media and newspaper advertisements. Then they gather contact details of relevant service provides and they may need to manually contact those service providers to make reservations and bookings. These methods may differ as some service providers has enabled web or social medial booking services while some are operating completely manually. The

payments process is also differed from one reservation to another. This leads to poor maintainability of the event budget according to specific event tasks.

1.5 Aims and Objectives

Proposed system, “Evenza” is an online event management and resourcing platform. This system as a platform where event planners or the customers by themselves could directly connect with the service provider or the product supplier for event related products and services resourcing,

The general objectives of system are to:

1. Enable user plan and management events online.
2. Enable resourcing event related services and products in one platform.
3. Enable event plan tackle and reminders for a smooth planning process.
4. Enable participant registration, invitation and management.
5. Provide a marketplace for event related service providers and product suppliers.
6. Provide progress evaluation and business supportive reports.

The specific objectives of the System are to:

1. Enable those planning for events to do so easily, effectively and efficiently.
2. Give customers an attractive, logical planning and resourcing experience.
3. Give service providers and suppliers a platform to advertise their goods and services online.
4. Enable event planners to monitor what they have ordered online and tackle their event plan.
5. Save time and other valuable resources for those planning for events.

1.6 Scope of the project

Proposed system is a platform based on resourcing event related service providers and product suppliers. The system is a web-based system which provides mobile interfaces for quick interaction with its users. Payments are collected through the system with aid of a payment gateway. Only an initial amount of each service cost or the payment quote will be charged through the system. Ordered product delivery is explicit of the system and will depend on the relevant delivery service provider assigned by the product supplier. The overall scope of the project is depicted by the figure 1.1 below.

Figure 1.0.1 Scope of the Project

1.7 Tools and Project Feasibility

1.7.1 Technical Feasibility

The system is planned to run on a cloud-based server (Web hosting) with the support of node js and express js and the front end will be developed using an interactive framework, Angular 8 and the database will be relational SQL database developed with MySql. The system will be designed to support concurrent access and smooth processing with the support of recent frameworks coding styles and patterns which enables easy maintenance and consistency.

1.7.2 Economical Feasibility

Because of new system implements with existing hardware sources and using opensource software's there will be no huge cost for deploying and running system. just only some minor neglectable costs. But with time faculty must bear up considerable cost in system maintenance

1.7.3 Operational Feasibility

Since the workload becomes less with the automating of event management process time consumption and the resource wastage on planning and finding required services will be reduced completely. And user Interfaces are very simple and interactive where event a user with basic IT knowledge can manipulate system easily.

notifications make their tasks efficient and remind them of their responsibilities in return providing operational benefits with reduction of cost and time used.

1.7.4 Organizational Feasibility

As an event management organization, the proposed system addressed in simplifying the tasks and process of a general event planner. So, the system can be widely used in such organizations in order to simplify their day to day tasks. Introducing the system through familiar technology will minimize the impact of change that the organizational employees have to prepare for.

1.8 Organization of the Report

This thesis covers the system analysis and design segments of the proposed system for the online event management and resourcing platform “Evenza”.

Chapter 1 introduces the current business process, problems and the objectives of the proposed system. It clearly defines the scope and the boundaries.

Chapter 2 presents the clear analysis of the existing system and the requirement analysis of the system. It further analyses the available business system options (BSOs) and defines the best option to proceed with.

Chapter 3 is the depiction of the furtherance of the project after the requirement analysis and specification. This will provide a better understanding of the system behavior and the interactions with the diagrams used. One diagram will lead to another to explain the functionality, entries and their relationships. By the end of the chapter, database design will show the tables which will be used in the system and Graphical User Interfaces will further elaborate its performance.

Chapter 4 is the conclusion of the report.The chapter will conclude the system analysis and design phases with a discussion of provided functionalities of thee system , benefits, drawbacks and the challenges ahead in development with suggestions.

1.9 Summary

In Chapter 1, the nature of the business process and the current functionality of the manual process of event organization, resourcing and management were identified. Then the problems and issues were defined, through which the objectives were emphasized. Furthermore, clear boundaries and scope for the system were defined.

CHAPTER 2

2 SYSTEM ANALYSIS

This chapter briefly analyses the current functionality of the existing business process and functional and nonfunctional requirements that are going to implement with the proposed system. It describes the requirement gathering techniques, how data are analyzed and the requirement definitions. Then going into much deeper, it discusses what are the Business System Options (BSO) available for the proposed system and finally it decides the best BSO to design the system.

2.1 Requirement Analysis

Gathering and analysis of requirements of the current business process of event management and resourcing is a must to develop a system with more usability and satisfaction. The requirement gathering was done by conducting interviews with event resourcing organizations as ‘Sarasi hotels- Matara’, ‘Evuni Saloon - Matara’, ‘Veura Photography’ etc. Over-the-phone interview was also conducted with Mr.Pasindu de Silva, (Biz lead - Ideamart for Dialog Axiata PLC) on technical and IT event management and resourcing in Sri Lanka. A background research was conducted on currently available event management organizations like ‘Magical Moments wedding planners’, ‘The Wedding Planner’ mobile application, ‘Block Events’ , ‘Meetup.com’, ‘RSVP.lk’, ‘OpenEvents’ and ‘eventray’ from APACHE and also with already established event management and coordinating organizations like 3N Events, Vibes events, Beyond Boundaries Pvt. Ltd, Imagine Entertainment and Showtown entertainment.

2.2 Use Case Diagram for the Current System

The functionality of existing business process of event planning, management and resourcing is handled manually and semi-autonomous approach with the aid of some currently available event task scheduling tools and participant management platforms. An exact boundary for the manual process cannot be clearly identified as it is open for all users who involve in this process under different levels. Therefore, the consistency of the process is very poor.

Figure 2.1 shows the overall scope of the existing process which carries out by an event planner (a firm/person required to organize an event) manually with use of available tools.

Figure 2.1 UC10 overall use case diagram for current business process

Figure 2.1 illustrates how the users interact with current business processes such as planning events and allocating budget according to the plan, purchasing required products, scheduling visit appointments, booking venue and services, get participants registered and manage registered participants. Current process is operated manually, and resources are gathered over external contacts and recommendations. Creating event plans, tracking plan and participant management are also operated manually with aid of a set of already available tools by different sources as listed below.

- Event planners and reminder applications (Tasks should be set manually, need to place reminders on added tasks)
- Online participant management platforms (Get participants registered to the event)
- With the support of an event planner or an event coordinator. (Or an event planning/organizing firm)

The user A10 - Customer has two different roles the customer(organizer) by themselves or a hired event planner. If the customer wants to plan and resource an event, can proceed directly according to the use case. If the customer hires an event planner or a coordinator to plan the event, then the event planner or coordinator will become the A10 -Customer to the system and they are required to go through the process according to the given use case. Event related service and product suppliers involved in the system are considered as the user 'A20 -Merchant'. Event participants who explicitly connected to the system are identified as the user 'A30 - Participant'.

2.3 Activity Diagrams for the Current System

2.3.1 Activity Diagram for creating an event plan

Figure 2.2 illustrates the activity diagram for creating an event plan. The activity is done by the customer (event planner). The current process is partially manual and achieved through the aid of different resources.

Figure 2.2 Activity Diagram for creating an event plan

2.3.2 Activity Diagram for finding required service providers

Figure 2.3 illustrates the activity diagram for finding required service providers. The activity is done by the customer (event planner).

Figure 2.3 Activity Diagram for finding required service providers

2.3.3 Activity Diagram for finding required product suppliers

Figure 2.4 illustrates the activity diagram for finding required product suppliers. The activity is done by the customer (event planner). As illustrated the required products are first identified and then the suitable product suppliers are found.

Figure 2.4 Activity Diagram for Finding required product suppliers

2.3.4 Activity Diagram for booking a service

Figure 2.5 illustrates the activity diagram for booking an event related service. The activity is done by the customer (event planner). Service can be either a venue or other service which is booked for a specific date and a time duration at the event day or prior to the event.

Figure 2.5 Activity Diagram for booking a service

2.3.5 Activity Diagram for create a visit appointment for a venue service

Figure 2.6 illustrates the activity diagram for creating a visit appointment for a venue service. The activity is done by the customer (event planner). This activity is identified when the customer needs to visit the venue (auditorium, reception hall etc..) before proceeding a booking.

Figure 2.6 Activity Diagram for create a visit appointment for a venue service

2.3.6 Activity Diagram for purchase products for the event

Figure 2.7 illustrates the activity diagram for purchasing required products for the event. The activity is done by the customer (event planner). Some products can be ordered online but most of the product suppliers do offline business.

Figure 2.7 Activity Diagram for purchase required products for the event

2.3.7 Activity Diagram for inviting and managing event participants

Figure 2.8 illustrates the activity diagram for inviting and managing registered participants for an event. The activity is done by the customer (event planner)

Figure 2.8 Activity Diagram for inviting and managing event participants

2.4 System Requirement Analysis

System requirements analysis phase aims at providing a full description of the problem based on the concepts defined in the problem domain. This section would brief the reader of the functional and non-functional requirements identified in the system analysis phase and those were identified as mandatory requirements with the wording 'shall' and non-mandatory requirements with the wording 'should'.

2.4.1 Functional Requirements

The table 2.1 will illustrate the functional requirements of the system.

Table 2.1 Functional requirements

ID	Requirement
1	Shall be able to create and manage user accounts.
2	Shall be able to login to the system.
3	Shall be able to create business profiles (by service providers)
4	Shall be able to manage products/services (by service providers)
5	Shall be able to provide a set of event types to select.
6	Shall be able to get event details for selected event type from customers.
7	Shall be able to generate an event plan for selected event type.
8	Shall be able to search for products/services under specific event plan segments.
9	Shall be able to book/request appointments for services (by customers)
10	Shall be able to order products (by customers)
11	Shall be able to send cancellation/ modification requests for bookings/appointments (by customers)
12	Shall be able to respond for bookings/appointments (by service providers)
13	Shall be able to provide appointment/booking calendar for service providers.
14	Shall be able to record booking and purchase history.
15	Shall be able to show notifications related to event plan.
16	Shall be able to record event plan details.

17	Shall be able to get customer feedback.
18	Shall be able to generate reports.
19	Shall be able to send invitations to event participants.
20	Shall be able to send event related notifications to participants.
21	Should be able to get participants registered to the relevant event.
22	Shall be able to backup data.
23	Should be able to manage registered users.
24	Should be able to receive payments (subscription fee and commission).

The functional requirements described in table 2.1 are what the system should be able to do in order to satisfy its main objectives. Mainly these describe what are the tasks or functions which the system can perform after initial implementation.

2.4.2 Non-functional Requirements

Non-functional requirements describe the behavior of the system other than the main functionality of it. Hence, they cover the requirements which are not included in section 2.4.1. The table 2.2 will show the non-functional requirements identified.

Table 2.2 Non - functional requirements

ID	Requirement
1	Shall be able to provide web interface.
2	Shall be able to provide mobile interface.
3	Shall be able to facilitate concurrent access.
4	Should be able to provide chatbot service to get event details and generate event plans.
5	Should be able to have a user-friendly attractive interface.
6	Should be able to share created events through social media.
7	Should be able to have quicker response time.
8	Should be able to provide support documentation and accessibility features.
9	Should be able to use the existing resources efficiently.

The non-functional requirements in table 2.2 describe the usability, reliability, performance, maintainability and other similar aspects of the system. These set of requirements may not be directly related to the main functionality, but they are of extreme importance to the proper functioning of the system.

2.4.3 Functional Requirement explanation

This section explains the functionalities that are going to be computerized in the proposed system. All requirements are organized to provide a full review of the system and to describe them in order.

1. Shall be able to create and manage user accounts.

- Shall be able to facilitate customer account creation and management.
- Shall be able to facilitate business owner/service provider account creation and management.
- Shall be able to facilitate administrator account creation and management.

2. Shall be able to login to the system

- Shall be able to facilitate different access levels for each account type.
- Shall be able to reset passwords.

3. Shall be able to create business profiles (by service providers)

- Shall be able to add details, business location and photos to the business profile
- Shall be able to create business profiles in two main types (service providers (venue and other services) and sellers(products))
- Shall be able to manage details in the business profile

4. Shall be able to manage products/services (by service providers)

- Shall be able to add new products/ services

- Shall be able to modify products/ services
- Shall be able to manage service availability status
- Shall be able to manage remaining product inventory

5. Shall be able to provide a set of event types to select

6. Shall be able to get event details for selected event type from customers

    - Shall be able to get event details
    - Shall be able to get planned event location
    - Shall be able to get planned event budget

7. Shall be able to generate an event plan for selected event type

    - Shall be able to generate plans for provided event types
    - Shall be able to create new event types and plans
    - Shall be able to allocate given budget for each plan segment
    - Shall be able to manage and customize event plan segments

8. Shall be able to search for products/services under specific event plan segments

    - Shall be able to provide services/products list on their availability
    - Shall be able to filter services/products on budget and given details
    - Shall be able to view user ratings and reviews
    - Should be able to filter services/products on event location

9. Shall be able to book/request appointments for services (by customers)

    - Shall be able to request visit appointments for venue providers
    - Shall be able to send booking requests for services

- Shall be able to pay online for confirmed bookings

10. Shall be able to order products (by customers)

- Shall be able to pay online for products

11. Shall be able to send cancellation/ modification requests for bookings/appointments (by customers)

12. Shall be able to respond for bookings/appointments (by service providers)

- Shall able to receive service booking/visit appointment requests
- Shall be able respond to booking/appointment requests
- Shall be able to modify bookings/appointments

13. Shall be able to provide appointment/booking calendar for service providers

- Shall be able to add bookings/appointments to the calendar
- Shall be able to consider booking availability on calendar when listing search results
- Should be able to update calendar with other bookings explicit to the system

14. Shall be able to record booking and purchase history

- Shall be able to record service bookings
- Shall be able to record visit appointments
- Shall be able to record product purchase history

15. Shall be able to show notifications related to event plan

- Shall be able to receive reminders
- Shall be able to receive event related notifications

16. Shall be able to record event plan details

17. Shall be able to get customer feedback

- Shall be able to rate and review services/products
- Shall be able to rate and review service providers

18. Shall be able to generate reports

- Shall be able generate sales reports
- Shall be able to generate payment records
- Shall be able to generate related performance measurement reports

19. Shall be able to send invitations to event participants

- Shall be able to send invitations via emails
- Shall be able to view no. of participants
- Should be able to view RSVP responds

20. Shall be able to send event related notifications to participants

- Shall be able to send event related notification via email
- Should be able to send event reminders via email

21. Should be able to get participants registered to the relevant event

- Should be able to send registration forms
- Should be able to view registered participant details

22. Shall be able to backup data

23. Should be able to manage registered users

24. Should be able to receive payments (subscription fee and commission)

2.5 Business System Options

The project needs to be focused on satisfying the identified requirements and future requirements that can be emerged in future. According the above requirement analysis and specification the system should mainly satisfy the customer's need to plan the relevant event according to the budget, find required venue, service providers and book services, order goods, keep a track on the event plan and get event participants registered.

In this section BSOs are presented and at the end they are evaluated to select the best option. Each BSO is consisted of overlapping features as well as exclusive features. Since each BSO is focused on different aspects, each BSO requires different set of software combinations. Evaluation of the BSOs are done by comparing them against the functional and non-functional requirements of the system.

2.5.1 Business system option identification

BSO1 – Web based event management system with Progressive mobile web application (PWA)

BS02 – Web based system with android and iOS based native mobile applications

BS03 - Web based application which utilize an existing event planning platform using its API (ex: - eventray, openevent)

2.5.2 BSO1- Web based event management and resourcing system with Progressive mobile web application (PWA)

2.5.2.1 Description

This BSO is concerned of developing a web-based event management and resourcing platform with all the features and develop it further to a progressive web application (PWA) with significant access to mobile devices as well (geo location, auto-update etc.). The system is platform independent and can be accessed anytime anywhere along with using any mobile device or desktop environment which has an internet access.

2.5.2.2 Functionality

Users can create their profiles and are provided more secure login. Customers can select an event type an generate a customizable event plan and select between suggested resources (service providers and sellers). Merchants gain an easy and convenient way of managing their business pages and add services/products. Once an event is created user is be able to manage its participants easily. Database including user data are stored in a cloud-based hosting service for concurrent reliable access. System provides highly interactive and common interface in both web and mobile devices.

2.5.2.3 Benefits

- Can be accessed anywhere through internet.
- Can be accessed even by using a smart phone or tablet computer with the feel of a native mobile application at anywhere anytime.
- Provide more interactive user interface which makes users comfortable with.
- Platform independent. (web, android mobile, IOS mobile)
- Accessible to mobile device features as location access etc. with the same platform.
- Provide generic interface for both web and mobile platforms.
- Data transfer with backend is gone through a proper validation and use of API.

2.5.2.4 Issues

- Rely on good internet connection.
- Security issues.

2.5.2.5 Planned Cost

Table 2.3 Planned cost - BSO1

Description	Cost(Rs)
Software (web based) development	100000
Mobile interface development	40000
Web domain	4000
Hosting and other cloud services	6000
Purchase of development tools	2000
Sub Total	**152000**

2.5.2.6 Justification

This BSO satisfy all the functional requirements and non-functional requirements of the proposed system. Since this is an online solution this gives flexibility for both customer and merchant as it can be accessed from anytime and anywhere. Cost is comparatively low, and the development time is minimum. Both web application and a mobile application can be provided to the user for wider use of the system. And this BSO also supplies more interactive interfaces.

2.5.3 BS02 – Web based system with android and iOS based native mobile applications

2.5.3.1 Description

This BSO is focused on implementing a web-based system with native mobile applications for event management and resourcing which is to be developed for both android and iOS platforms. This BSO supplies a native mobile application of the proposed system and can be accessed from anywhere anytime along with good internet connection.

2.5.3.2 Functionality

Both customers and the merchants need to create user accounts and access to the system using a web browser or a mobile device with relevant login credentials. A natively developed mobile application is more advanced but the interfaces are limited to a small space and the multi-functionality may be difficult. Users may have to install an integrated application either from the 'Google play store' or 'Apple app store' to use the system and it will reserve more space on the device.

2.5.3.3 Benefits

- Can be access from anywhere through internet even while travelling.
- Easy navigation through the system.
- Direct access to device features.
- More interactive.

2.5.3.4 Issues

- High development and deployment cost due to building a separated native mobile application for both android and iOS platforms.
- Users must have a proper mobile device to access the application (ex: Suitable android OS and memory for a smooth run of the application).
- Devices with lower android versions may not able to access the application.
- Takes additional space on mobile devices.

2.5.3.5 Planned Cost

Table 2.4 Planned cost - BSO2

Description	Cost (Rs)
Web application development	100000
Mobile application development	80000
Play Store and App Store payments(deploying)	7000
Hosting and other cloud services	6000
Purchase of development tools	3000
Sub Total	**196000**

2.5.3.6 Justification

This BSO satisfied the functional requirements and many non-functional requirements of the proposed system. But there are a few drawbacks with this BSO. The mobile application has limited access for devices for previous android versions, and the additional space reserved on the device, limited interfaces may not capable of providing all the information and mufti-functionality. High development and deployment cost as well as the increased development time are other drawbacks.

2.5.4 BSO3 - Web based application which utilize an existing event planning platform using its API (Ex: - eventray, openevent)

2.5.4.1 Description

In this BSO developing a web-based application integrated with the support of an already available event management API (Ex: - open-source platforms like eventray, openevent by APACHE) is concerned. A community can be developed as a part of this implementation and data mining techniques can be further used for more productive event plan generation and suggesting. But the system may require separate APIs from different platforms for event management and resourcing features separately as the currently available platforms does not provide both services together. Some required functionalities may not be able to provide using currently available APIs.

2.5.4.2 Functionality

All users must create profiles and user data are stored and managed in the relevant event management platform as part of it. The provided API services are used. Other functionalities are same as the previous BSOs.

2.5.4.3 Benefits

- Enables wide data availability for data mining purposes
- Quick response time
- High concurrent access
- Easy data storage and management
- Cloud storage facilities

2.5.4.4 Issues

- Incompatibility
- Data security and privacy concerns
- Need to use different APIs with separated data sources
- Inability to provide some functionalities.

2.5.4.5 Planned Cost

Table 2.5 planned cost - BSO3

Description	Cost (Rs)
Web based development	100000
Purchase of API Services	70000
Hosting and other cloud services	5000
Purchase of development tools	3000
Sub Total	**178000**

2.5.4.6 Justification

Though this BSO is easy to implement and more secure way. But the main objective of this system is to provide a vendor-based platform to connect customer with merchants who provide event related resources. No currently available platform provides event resourcing, event management and participant management together through their APIs. There this BSO is not be the most appropriate one.

2.6 Evaluation of BSOs

Evaluation of the BSOs were done by comparing them first with the functional and non-functional requirements. Then the pros and cons of the BSOs were considered. Ultimately the best option could be one of the BSOs. Looking at the three BSOs, it seems that all of them cover almost all the requirements mentioned in the requirements catalogue. But there are some differences when taking the other constraints into consideration. The figure 2.6 shows the comparison of the each BSO with the requirements.

2.6.1 Functional requirements Vs. BSOs

Table 2.6 illustrates the comparison of the three BSO with the functional requirements. According the above comparison, all the functional Requirements are satisfied by the BSO1, BSO2 and BSO3.

Table 2.6 Functional requirements vs. BSOs

ID	Requirement	BSO1	BSO2	BSO3
1	Shall be able to create and manage user accounts.	X	X	X
2	Shall be able to login to the system.	X	X	X
3	Shall be able to create business profiles.	X	X	X
4	Shall be able to manage products/services (by service providers)	X	X	X
5	Shall be able to provide a set of event types to select.	X	X	X
6	Shall be able to get event details for selected event type from customers.	X	X	X
7	Shall be able to generate an event plan for selected event type.	X	X	X
8	Shall be able to search for products/services under specific event plan segments.	X	X	X
9	Shall be able to book/request appointments for services	X	X	X
10	Shall be able to order products	X	X	X

11	Shall be able to send cancellation/ modification requests for bookings/appointments	X	X	X
12	Shall be able to respond for bookings/appointments	X	X	X
13	Shall be able to provide appointment/booking calendar for service providers.	X	X	X
14	Shall be able to record booking and purchase history.	X	X	X
15	Shall be able to show notifications related to event plan.	X	X	X
16	Shall be able to record event plan details.	X	X	X
17	Shall be able to get customer feedback.	X	X	X
18	Shall be able to generate reports.	X	X	X
19	Shall be able to send invitations to event participants.	X	X	X
20	Shall be able to send event related notifications to participants.	X	X	X
21	Should be able to get participants registered to the relevant event.	X	X	X
22	Shall be able to backup data.	X	X	X
23	Should be able to manage registered users.	X	X	X
24	Should be able to receive payments (subscription fee and commission).	X	X	X

2.6.2 Non-functional requirements Vs. BSOs

Table 2.7 illustrates the comparison of the three BSOs with the non-functional requirements. According the above comparison, all the non-functional requirements are satisfied by the BSO1. BSO2 and BSO3 are satisfying only a few non-functional requirements.

Table 2.7 Non-functional requirements vs. BSOs

ID	Requirement	BSO1	BSO2	BSO3
1	Shall be able to provide web interface.	X	X	X
2	Shall be able to provide mobile interface.	X	X	X
3	Shall be able to facilitate concurrent access.	X	X	X
4	Should be able to provide chatbot service to get event details and generate event plans.	X	X	
5	Should be able to have a user-friendly attractive interface.	X	X	
6	Should be able to share created events through social media.	X	X	X
7	Should be able to have quicker response time.	X		
8	Should be able to provide support documentation and accessibility features.	X		
9	Should be able to use the existing resources efficiently.	X		

2.6.3 Selected BSO with a sound justification

Selected BSO is BSO1- 'Web based event management and resourcing system with progressive mobile web application (PWA)' as it satisfies all the functional and non-functional requirements. It can provide more interactive and convenience use of the system while giving a more security and keeping customers and merchants connected as well as implementing a platform for easy management of events and its participants. It also provides the most affordable solution among all the stated BSOs with the minimum cost of Rs.152 000/- while providing more features. It can generate

managerial reports related to merchants and included a chatbot to make the use of system more convenience.

Table 2.8 BSO Justification

	BSO 1	**BSO 2**	**BSO 3**
Initial cost	Low	High	Medium
Security risk	Medium	Medium	High
Functional Requirements	High	High	High
Non - functional Requirements	High	Medium	Low
Variable cost	Medium	Medium	Medium

2.7 Summary

This chapter analyzed the existing features of the current business process and thus specified the functional and non-functional requirements that are about to be implemented in the proposed system. This evaluated three BSOs and finally selected 'Web based event management and resourcing system with a progressive mobile web application (PWA)' as the most appropriate Business System Option.

CHAPTER 3

3 SYSTEM DESIGN

This chapter mainly covers the design of the system. It describes the continuation of the project after the requirement analysis and specification. It gives a better understanding of the system functionalities, it's actors and behaviors and interactions with the diagrams used. One diagram leads to another with explaining the exact the functionality, entities and their relationships.

3.1 Use Case Diagram for the Proposed System

Use case provides a structured view of the system functionality. These diagrams are used to gather the requirements of a system, including internal and external requirements. These requirements are mostly design requirements. When a system is analyzed to gather its functionality, use cases are prepared and actors of the system are identified. This process is very important since the next phase of the System Development Life Cycle (SDLC) is the system designing phase. Here the use cases are the different tasks the users do in order to interact with the system. Actors are the users who will be interacting with the system.

The figure 3.1 illustrates the overall use case diagram for the proposed system. Event planners (or the customer) is denoted as 'A-10 Customer' and the merchant who provide event related services and products are denoted as 'A-20 Merchant' in two types 'A22- service Provider' and 'A-24 Product Supplier'. Events are categorized into two types as 'Open events' which, any random participant who satisfies given set of requirements through a registration, can be participated and 'Closed type' events that only an identified set of guests are invited through an email. Participants are also categorized regarding the above event categorization as 'A42- Invitee' for closed type events and 'A44- Random participant' for open type events.
Any product can be purchased, and any service can be booked while a visit appointment can be appointed only for 'venue' type services.

Figure 3.0.1 UC20 Overall use case for the proposed system

3.1.1 UC92 Use Case for update user profile

Figure 3.2 illustrates the use case diagram for update user profile. The activities are done by both customer and Merchant.

Figure 3.2 UC92 Use Case for update user profile

3.1.2 UC110 Use Case for create an event plan

Figure 3.3 illustrates the use case diagram for creating an event plan using the system. The activities are done by the customer.

Figure 3.3 UC110 Use Case for create event plan

3.1.3 UC120 Use Case for search and sort products or services

Figure 3.4 illustrates the use case diagram for search and sort event related products and services. The activities are done by the customer.

Figure 3.4 UC120 Use Case for search and sort products or services

3.1.4 UC130 Use Case for create business profile

Figure 3.5 illustrates the use case diagram for create business profile. The activities are done by the merchant.

Figure 3.5 UC130 Use Case for create business profile

3.1.5 UC140 Use Case for add products and services

Figure 3.6 illustrates the use case diagram for add services and products to the business page.

Figure 3.6 UC140 Use Case for add products and services

3.1.6 UC150 Use Case for place a booking for a service

Figure 3.7 illustrates the use case diagram for placing a booking for a service.

Figure 3.7 UC150 Use Case for place a booking for a service

3.1.7 UC160 Use Case for place a visit appointment for a service

Figure 3.8 illustrates the use case diagram for place a visit appointment for a venue.

Figure 3.8 UC160 Use Case for place a visit appointment for a service

3.1.8 UC170 Use Case for place an order for a required product

Figure 3.9 illustrates the use case diagram for place an order for a event related product. Payment can be made online.

Figure 3.9 UC170 Use Case for place an order for a required product

3.1.9 UC180 Use Case for send invitations to the 'closed type' event invitees

Figure 3.10 illustrates the use case diagram for send invitations to the 'closed type' event participants.

Figure 3.10 UC180 Use Case for send invitations to the closed type event invitees

3.1.10 UC190 Use Case for get participants registered for the 'open type' events

Figure 3.11 illustrates the use case diagram for get participants registered for 'open type' events through registration.

Figure 3.11 UC190 Use Case for get random participants registered to the open type events

3.1.11 UC200 Use Case for generate reports

Figure 3.12 illustrates the use case diagram for generate reports.

Figure 3.12 UC200 Use Case Diagram for generate reports

3.1.12 UC210 Use Case for collect commission and subscription fee by System Administrator

Figure 3.13 illustrates the use case diagram for collect commission and subscription fee by System Administrator.

Figure 3.13 UC210 Use Case for collect commission and subscription fee by System Administrator

3.1.13 UC220 Use Case for rate and review services/ products

Figure 3.14 illustrates the use case diagram for rate and review service and products by the customer.

Figure 3.14 UC170 Use Case for rate and review services/ products

3.2 Use Case Descriptions for the Proposed System

3.2.1 UC92 Use case description for update user profile

Table 3.1 UC92 Use case description for update user profile

Use Case	UC92 - Update user profile
Use Case Description	This use case describes how the user can update details in their user profile.
Actors	A 10 - Customer, A20 -Merchant
Pre-conditions	User should have logged in to the system.
Basic Flow	1.Navigate to 'User Profile' 2.Update profile info 3.Add a profile picture 4. Check user type [if a merchant] 4.1 confirm identity verification if not confirmed yet 5.User may change the password [if password change required] 5.1 Type old password 5.2 Type new password
Alternative Flow/ Exceptions	1. Entered data may not be validated and given error. 2. Uploaded picture may be larger than the given size limit.
Post-conditions	Successfully logged in to the system and given desired access level

3.2.2 UC110 Use case description for create an event plan

Table 3.2 UC110 Use case description for create an event plan

Use Case	UC110 - Create an event plan
Use Case Description	This use case describes how the user can create a new event plan for already available event type or newly created event type.
Actors	A 10 - Customer
Pre-conditions	User should have logged in to the system.
Basic Flow	1.Navigate to 'Event'. 2.Select 'Create new Event'. 3.Select event category. [if category not available] 3.1 Create a new event category. 4. Select event type (Open event/closed event) 5. Input required event details. 6. Validate given details [if valid] 6.1 generate event plan with a set of plan segments [if not valid] 6.2 Show error. request to try again.
Alternative Flow/ Exceptions	1.Entered data may not be validated and given error.
Post-conditions	Creates an event plan with set of segments.

3.2.3 UC120 Use case description for search and sort services or products

Table 3.3 UC120 Use case description for search and sort services or products

Use Case	UC120 - Search and sort services or products
Use Case Description	This use case describes how the user can search for specific services and/or products for the event.
Actors	A 10 - Customer
Pre-conditions	1.User should have logged in the system. 2.Merchant should have added products or services to the system. 3.User should have created an event plan or navigated to the general search area.
Basic Flow	1.User click on 'Search' and type a keyword or select an event plan segment which is a type of 'Service' or 'Product' 2.System generates list of suggested services or products according to provided filters 3.User filter results by price, user ratings, service provider location and other given filters.
Alternative Flow/ Exceptions	1.Entered keyword may not match with any available resources. 2.Event plan may have not yet generated. In that case user first need to generate an event plan according to the event.
Post-conditions	The most suitable services products are suggested. User will view and book/purchase services and/or products.

3.2.4 UC130 Use case description for create a business profile

Table 3.4 UC130 Use case description for create a business profile

Use Case	UC130 - Create a business profile
Use Case Description	This use case describes how the merchant can create a business profile.
Actors	A20 -Merchant
Pre-conditions	Merchant should have logged in to the system. Merchant should have completed the identity verification in their user profile.
Basic Flow	1.User click on 'Create Business Profile' 2.Check for Identity Verification [if verified] create a blank business profile. [if not verified] Show an error. Request to verify first. 1. Update business title, type, category and description. 2. Update business location, address and contact details. 3. Upload photos related to the business. 4. Upload any available certification and earn a badge after verification of uploaded certification. 5. UC 140 - Add products or services to the business profile. 6. Validate profile details [if valid] 8.1 Update business profile details. [if not valid] 8.2 Show an error.
Alternative Flow/ Exceptions	1.Details my have not provided correctly 2.Identity verification may have not completed
Post-conditions	A business profile is created and published.

3.2.5 UC140 Use case description for add services or products to the business profile

Table 3.5 UC140 Use case description for add services or products to the business profile

Use Case	UC140 - Add services or products to the business profile
Use Case Description	This use case describes how the user merchant can add products or services to the business profile, to be shown to the customers for booking / purchasing.
Actors	A20 -Merchant
Pre-conditions	Merchant should have logged in to the system. Merchant should have created and published a business profile.
Basic Flow	1.User click on 'Add service / Add product' 2.Select service or product category 3.Input title, description and other details. 4. Add photos of the service or product. 5.Set product/service state [if a service] 5.1 Set service availability [if a product] 5.1 Set product inventory 6.Validate provided details [if valid] 6.1 Created and update service or product 6.2 Add service or product to the business profile [if not valid] 6.3 Show an error
Alternative Flow/ Exceptions	
Post-conditions	Successfully added service or product to the business profile. User will receive a confirmation message.

3.2.6 UC150 Use case description for place a booking for a service

Table 3.6 UC150 Use case description for place a booking for a service

Use Case	UC150 - Place a booking for a service
Use Case Description	This use case describes how the user can place a booking for a selected service.
Actors	A10 - Customer, A22 - Service Provider, A50 -Payment gateway
Pre-conditions	Customer should have logged in to the system. An event plan should be generated and selected.
Basic Flow	1.User select relevant event plan segment for a service. 2.A suggested list according to the availability, event location and allocated budget, will be generated. 3.User can change the required date, time and filter again. 4.User can sort list with provided filters. 5.Select a suitable service in the list. 6.Select 'Book Service' [if date is available] [if can be directly booked] 6.1.1 check calendar availability and reserve booking. 6.1.2 generate payment quote and send to the customer. [if cannot be directly booked] 6.1.3 A booking request will be sent to the service provider. 6.1.4 Booking confirmed by the service provider. 6.1.5Payment quote will be generated.

	[if date in unavailable] 6.2 Show error and request another selection. 7.User make payment through the payment gateway. (An initial amount of the total booking) 8.Booking will be confirmed, and payment receipt will be generated. 9.Booking calendar will be updated, and history will be recorded.
Alternative Flow/ Exceptions	1.If event plan segments relevant to required services are not generated, use can add a new plan segment and book a service. 2.User can directly search for a service and book it.
Post-conditions	A booking successfully created. Payment quote for booking reservation and the payment receipt once the payment is made will be sent to the customer.

3.2.7 UC160 Use case description for place a visit appointment for a venue service

Table 3.7 UC160 Use case description for place a visit appointment for a venue service

Use Case	UC160 - Place a visit appointment for a venue service
Use Case Description	This use case describes how the user place a visit appointment for a service under 'venue' category prior to pay and booking.
Actors	A10 - Customer, A22- Service Provider
Pre-conditions	Customer should have logged in to the system.
Basic Flow	1. User select relevant service under 'venue' category. 2. Select 'Create a visit appointment' 3. User can provide/modify the required date, time for the

	visit appointment. [if date is available] 3.1 check calendar availability and reserve appointment. 3.2 send appointment request to service provider. [if date in unavailable] 3.3 Show error and request another date/time. 4. A reservation note sent to the customer. 5. Booking calendar will be updated. 5. Appointment history will be recorded.
Alternative Flow/ Exceptions	1.If event plan segments relevant to required services are not generated, use can add a new plan segment for a venue service. 2.User can directly search for a service and create a visit appointment.
Post-conditions	A visit appointment is successfully scheduled. A reservation note will be sent to the customer.

3.2.8 UC170 Use case description for place an order for a required product

Table 3.8 UC170 Use case description for place an order for a required product

Use Case	UC170 - Place an order for a required product
Use Case Description	This use case describes how the user place an order for a required product
Actors	A10 - Customer, A24 - Product Supplier, A50 - Payment gateway
Pre-conditions	Customer should have logged in to the system.
Basic Flow	1. User select relevant product from the list. 2. View product details and price. 3. User select ‘order now’

	4. User provide required quantity. 5. Check for the availability in inventory [if required quantity is available] 5.1 reserve the quantity. 3.2 send payment quote to the user. [if not unavailable] 3.3 Show error and request another selection. 4. User make payment through the payment gateway. (including delivery charges if required) 5. Request will be sent to the product supplier. 6. Payment receipt will be generated and sent to the customer. 7. Inventory will be updated. 8. Order history will be recorded. 9. Products are overhanded to the delivery (explicit from the system) 10. Delivery service details and a delivery ID will be sent to the user.
Alternative Flow/ Exceptions	1.If event plan segments relevant to required products are not generated, use can add a new plan segment for a product category. 2.User can directly search for a product and place an order. 3. Delivery is explicit from the system and only the details will be provided. Delivery tracking will depend on the delivery service hired by the product supplier.
Post-conditions	A product order is successfully created. A reservation note will be sent to the customer.

3.2.9 UC180 Use case description for send invitations to the closed type event invitees

Table 3.9 UC180 Use case description for send invitations to the closed type event invitees

Use Case	UC180 - Send invitations to closed type event invitees
Use Case Description	This use case describes how the user send invitations for a closed type event to a set of selected invitees.
Actors	A10 - Customer, A42 - Invitee
Pre-conditions	Customer should have logged in to the system. Customer should have created a closed type event. The event should have been in 'published' state.
Basic Flow	2. User select an event of closed type. 2. select 'Participants'. 3. View already confirmed invitees and number of guests. 4. select 'Create Invitation' 5. Create invitation content and save draft. 6. select 'Add new Participant' 7. Input participant email. 8. send invitation. 9. Invitee receive and confirm invitation. [if confirmed] 9.1 update the participant list. 9.2 send upcoming event related notifications to the invitee.
Alternative Flow/ Exceptions	1.Once the event is in published invitation can be share over social media if the email of the invitee is not available.
Post-conditions	An invitation for an event is created. Participant is added to the event.

3.2.10 UC190 Use case description for get random participants registered to the open type events

Table 3.10 UC190 Use case description for get random participants registered to the open type events

Use Case	UC190 - Get random participants registered to the open type events.
Use Case Description	This use case describes how the user get random participants registered to an open type event.
Actors	A10 - Customer, A44 - Random Participant
Pre-conditions	Customer should have logged in to the system. Customer should have created an open type event. The event should have be in 'published' state.
Basic Flow	1. User select an event of open type. 2. select 'Participants'. 3. View already confirmed participants and number of guests. 4. select 'Create Registration form' 5. Create registration form and get the form link. 6. select 'Add new Participant' 7. send registration form link through email. 8. Participant receive the registration form, fill and submit it. [if approved] 8.1 update the participant list. 8.2 send upcoming event related notifications to the participant.
Alternative Flow/ Exceptions	1.Once the event is in published invitation can be share over social media if the email of the invitee is not available.
Post-conditions	A registration form for an event is created. Participant is added to the event.

3.2.11 UC200 Use case description for generate reports

Table 3.11 UC200 Use case description for generate reports

Use Case	UC200 - Generate reports.
Use Case Description	This use case describes how the user generate different types of reports through the system.
Actors	A20 - Merchant, A30 - System Administrator
Pre-conditions	User should have logged in to the system.
Basic Flow	1. Navigate to 'Report' wizard. 2. select ''New Report'. 3. Select report subject (Sales/ bookings/ commission/ subscription etc.) 4. Select report type. 5. Provide duration to consider. 6. Add necessary filters. 7. select 'Generate Report' 8. View report and export as a PDF.
Alternative Flow/ Exceptions	-
Post-conditions	A report of evaluation is generated.

3.2.12 UC210 Use case description for collect commission and subscription fee by system administrator

Table 3.12 UC210 Use case description for collect commission and subscription fee by system administrator

Use Case	UC210 - Collect commission and subscription fee by system administrator.
Use Case Description	This use case describes how the system administrator collects commission on sales and bookings and the subscription fee.
Actors	A20 -Merchant, A 30 -System Administrator
Pre-conditions	System Administrator should have logged in to the system.
Basic Flow	1. Service Provider renew their subscription. 2. A commission is generated on a product order or a booking. 3. Subscription fee and commission amount forwarded to the system administrator through payment gateway. 4. Merchant's subscription renewed.
Alternative Flow/ Exceptions	
Post-conditions	Subscription fee and commission are collected.

3.2.13 UC220 Use case description for rate and review services and products

Table 3.13 UC220 Use case description for rate and review services and products

Use Case	UC220 - Rate and review services and products.
Use Case Description	This use case describes how the user rate and review a booked service or an ordered product through the system.
Actors	A10 - Customer
Pre-conditions	Customer should have logged in to the system.
Basic Flow	1. Customer select a service or a product from the booking / order history. 2. customer will be immediately ask to rate and review a product or service as an option at the end of the each service booking and product purchase. 3. Provide a star rating (1-5) and a review if interested. 4. Save ratings and reviews.
Alternative Flow/ Exceptions	
Post-conditions	Rating and reviews will be recorded and published.

3.3 Activity Diagrams for the Proposed System

3.3.1 Activity diagram for sign up to the system

Figure 3.15 illustrates the activity diagram for sign up to the system. The activity is done by both customer and the merchant. Merchant can verify their identity while signing up or later.

Figure 3.15 Activity diagram for sign up to the system

3.3.2 Activity diagram for log in to the system

Figure 3.16 illustrates the activity diagram for log in to the system. The activity is done by customer, merchant and system administrator.

Figure 3.16 Activity diagram for log in to the system

3.3.3 Activity diagram for update Merchant's user profile

Figure 3.17 illustrates the activity diagram for update Merchant's user profile. The activity is done by merchant. Both merchant and customer can update their user profiles. Merchant's profile update includes identity verification with a photograph of NIC/DL. Without proper identity verification, merchant is not allowed to create a business profile.

Figure 3.17 Activity diagram for update Merchant's user profile

3.3.4 Activity diagram for generate event plan for available event category

Figure 3.18 illustrates the activity diagram for generate event plan for an available event category in the system. The activity is done by customer. An event has two types: open and closed, and it has several categories. An event plan will be generated accordingly.

Figure 3.18 Activity diagram for generate event plan for available event category

3.3.5 Activity diagram for and generate event plan for new event category

Figure 3.19 illustrates the activity diagram for generate event plan for new event category in the system. The activity is done by customer. A new event category can be created, and a basic plan can be generated after providing relevant details.

Figure 3.19 Activity diagram for and generate event plan for new event category

3.3.6 Activity diagram for creating business profile

Figure 3.20 illustrates the activity diagram for creating business profile. The activity is done by merchant. Merchant's identity verification is required. Products or services can be added to the merchant's business profile.

Figure 3.20 Activity diagram for creating business profile

3.3.7 Activity diagram for add and modify products and services

Figure 3.21 illustrates the activity diagram for add and modify products and services to the merchant's business profile. The activity is done by merchant.

Figure 3.21 Activity diagram for add and modify products and services

3.3.8 Activity diagram for search and filter products and services

Figure 3.22 illustrates the activity diagram for search and filter services or products. The activity is done by customer. They can either search for product or service or generate a suggested product/ service list under relevant event segment.

Figure 3.22 Activity diagram for search and filter products and services

3.3.9 Activity diagram for place a booking for a service

Figure 3.23 illustrates the activity diagram for place a booking for a service. The activity is done by customer. An initial booking fee will be charged through the system.

Figure 3.23 Activity diagram for place a booking for a service

3.3.10 Activity diagram for place a visit appointment for a venue service

Figure 3.24 illustrates the activity diagram for placing a visit appointment for a venue. The activity is done by customer. A visit appointment prior to the booking can be only requested for services belongs to 'Venue' category.

Figure 3.24 Activity diagram for place a visit appointment for a venue service

3.3.11 Activity diagram for place an order for a required product

Figure 3.25 illustrates the activity diagram for purchase and order products. The activity is done by customer. Ordering and payment can be processed through system. Product delivery is explicit from the system.

Figure 3.25 Activity diagram for place an order for a required product

3.3.12 Activity diagram for send invitations to invitees (closed type events)

Figure 3.26 illustrates the activity diagram for sending invitations for the selected guests for ‘closed type’ events. The activity is done by customer. RSVP details are recorded.

Figure 3.26 Activity diagram for send invitations to invitees (closed type events)

3.3.13 Activity diagram for get participants registered (open type events)

Figure 3.27 illustrates the activity diagram for getting participants registered to an event using a registration form. The activity is triggered by customer. A generic event registration form can be crated and send through the system.

Figure 3.27 Activity diagram for get participants registered (open type events)

3.3.14 Activity diagram for generating reports

Figure 3.28 illustrates the activity diagram for generating reports. The activity is done by merchant and the system administrator.

Figure 3.28 Activity diagram for generating reports

3.3.15 Activity diagram for collect commission by admin

Figure 3.29 illustrates the activity diagram for collecting commissions and subscription fee paid by the merchants. The activity is done by system administrator.

Figure 3.29 Activity diagram for collect commission by admin

3.3.16 Activity diagram for manage users by admin

Figure 3.30 illustrates the activity diagram for managing system users. Users are consisting of both customers and merchants. The activity is done by system administrator.

Figure 3.30 Activity diagram for manage users by admin

3.3.17 Activity diagram for chat with chatbot

Figure 3.31 illustrates the activity diagram for chat with chatbot, a feature which will be developed as a non-functional requirement. Customers can have a convenient chat with chatbot for easy event planning with less complexity.

Figure 3.31 Activity diagram for chat with chatbot

3.4 Sequence Diagrams for the Proposed System

Sequence diagrams show how the system interacts with the actors in a use case functionality. Each actor is represented with a horizontal lifeline and the data transactions are drawn from one lifeline to another or within one lifeline. Following sequence diagrams describe some of the main use cases which are a bit difficult to understand with only having use case descriptions.

3.4.1 Sequence diagram for sign up to the system

Figure 3.32 illustrates the sequence diagram for sign up to the system.

Figure 3.32 Sequence diagram for sign up to the system

3.4.2 Sequence diagram for log in to the system

Figure 3.33 illustrates the sequence diagram for log in to the system.

Figure 3.33 Sequence diagram for log in to the system

3.4.4 Sequence diagram for create business profile

Figure 3.34 illustrates the sequence diagram for creating the business profile by the merchant.

Figure 3.34 Sequence diagram for create business profile

3.4.5 Sequence diagram for create event plan

Figure 3.35 illustrates the sequence diagram for creating an event plan by customer.

Figure 3.35 Sequence diagram for create event plan

3.4.6 Sequence diagram for place a booking for a service

Figure 3.36 illustrates the sequence diagram for placing a booking for a service.

Figure 3.36 Sequence diagram for place a booking for a service

3.4.7 Sequence diagram for place a visit appointment for venue service

Figure 3.37 illustrates the sequence diagram for placing a visit appointment for a venue service prior to the booking.

Figure 3.37 Sequence diagram for place a visit appointment for venue service

3.4.8 Sequence diagram for order a product

Figure 3.38 illustrates the sequence diagram for order and purchase a required product for the event.

Figure 3.38 Sequence diagram for order a product

3.5 Class Diagram for Proposed System

Class diagram in figure 3.39 describes the structure of a proposed system by showing the system's classes and their attributes(properties), behaviors(methods) and the relationships among the classes. Attributes and methods are clearly identified as private and public, and static attributes, methods are also included. System includes object-oriented programming concepts as abstraction, inheritance and encapsulation.

Figure 3.39 Class Diagram for the proposed system

3.6 Database Design

3.6.1 Entity – Relationship model

The entity relationship diagram shown below in figure 3.40 describe the relationship between entities of the system including their degrees and dependencies.

Figure 3.40 Entity Relationship diagram for the proposed system

3.6.2 Normalized database diagram

Database relationship diagram shown in figure 3.41 illustrates the proposed data tables in the system their keys and the relationship between the data tables in the database.

Figure 3.41 Normalized database design for the proposed system

3.7 Proposed GUI Design

Graphical User Interface (GUI) is one of the key components when developing the system as it is important for the interactive and convenient use of customers and merchants within the system. The system designer must be concerned on developing user-friendly GUI which is able to calm and cool the mind of user as this is a system that expected to use by the user in day to day life. User friendly GUI is the one of the major non-functional requirements of this system. The main design consideration related with GUI's (Graphical User Interface) are listed below where these factors are considered in the system GUI designs.

- Attractive and user-friendly user interfaces.
- Easy to navigate forward and backward and keeping the flow of the actions.
- Give good error messages with relevant information to recover from the error occurred.
- Prevent errors as much as possible and use client-side validations to give feedback.
- Provide feedback of all the actions even if succeeded or not.
- More weight on process automation approach and reducing complexity through abstraction by designing smart interfaces.
- Try to implement and provide accessibility features for users with different requirements.

3.7.1 Welcome page GUI

Figure 3.42 Welcome page GUI

3.7.2 System Sign in GUI

Figure 3.43 System Sign in GUI

3.7.3 System Sign up GUIs (Merchant's sign up)

Figure 3.44 System Sign up GUIs (Merchant's sign up) 01

Figure 3.45 System Sign up GUIs (Merchant's sign up) - 02

Figure 3.46 System Sign up GUIs (Merchant's sign up) - 03

3.7.4 GUI of user dashboard

Figure 3.47 GUI of user dashboard

3.7.5 GUI of events overview

Figure 3.48 GUI of events overview

3.7.6 GUIs for creating an event plan

Figure 3.49 GUIs for creating an event plan 01

Figure 3.50 GUIs for creating an event plan 02

Figure 3.51 GUIs for creating an event plan 03

Total budget is allocated into separate plan segments. Reminders are set to each milestone. Customer can adjust and modify plan segments according to their requirements.

3.7.7 GUI of Merchant's business profile

Figure 3.52 GUI of Merchant's business profile

3.7.8 GUI of adding services to the business profile

Figure 3.53 GUI of adding services to the business profile

3.7.9 GUI of adding products to the business profile by a product supplier

Figure 3.54 GUI of adding products to the business profile by a product supplier

3.7.10 GUI of Service overview and booking

Figure 3.55 GUI of Service overview and booking

3.7.11 GUI of payment page

Figure 3.56 GUI of payment page

3.7.12 GUI of managing closed type event invitees

Figure 3.57 GUI of managing closed type event invitees

3.7.13 GUI of creating registration form for open type event participants

Figure 3.58 GUI of creating registration form for open type event participants

3.7.14 GUI of generating reports

Figure 3.59 GUI of generating reports

3.7.15 GUI of Bookings overview of a customer

Figure 3.60 GUI of Bookings overview of a customer

3.7.16 GUIs of mobile interface - Getting started page

Figure 3.61 GUIs of mobile interface - Getting started page

3.7.17 GUIs of mobile interface - Service list

Figure 3.62 GUIs of mobile interface - Service list

3.7.18 GUIs of mobile interface - sidebar navigation

Figure 3.63 GUIs of mobile interface - sidebar navigation

3.7.19 GUIs of mobile interface - chat with chatbot

Figure 3.64 GUIs of mobile interface - chat with chatbot

3.7.20 GUIs of mobile interface - Receive Notifications

Figure 3.65 GUIs of mobile interface - Receive Notifications

3.8 System Reports

The system provides different types of managerial reports and transactional documents aiming enabling sales predicting and forecasting for service providers and product suppliers(merchants), easy organization and management of payment collections for system administrator and convenient transactions through system for customers with providing relevant receipts and notes. Some reports are listed below.

Managerial Reports are generated as:

- Sales
- Service bookings
- Event budgeting
- Commission and subscription fee collections
- Forecasting for sales / service hiring
- Progress evaluation reports

Transactional documents are generated as:

- Payment quote
- Payment receipt
- Booking note
- Appointment reservation note
- Event invitation

3.9 Summery

This chapter depicted the system design where it showed how the functionality of the system is achieved with refer to several diagrams. This was depicted through objected oriented approach within designing its classes, their attributes and behaviors. Furthermore, the database design and the interactive graphical user interfaces were presented.

CHAPTER 4

4 SYSTEM DEVELOPMENT

This chapter depicts the information related to the development of the project. It provides a brief introduction to the technologies and the programming language properties used to implement the “Evenza” Online Event Management and Resourcing Platform. Furthermore, this chapter provides a brief description of the third-party components / libraries used and the data structures and algorithms used which is of significant value to the implementation process.

4.1 Programming languages

“Evenza” is a interactive web platform which caters the requirements of the event planners and the service providers. Evenza platform mainly focused on providing a marketplace for event related resource providers as well as providing event management and scheduling capabilities. Therefore, I used JavaScript based MEAN Stack: MongoDB as the database, Angular 8 as the frontend framework, Node JS and express framework as the backend to develop the system. Using MongoDB as the databased has enabled large scale, unstructured data handling, which is required for the application functionality. Angular frontend framework has enabled efficient data biding, DOM manipulation, attractive User Interfaces and Validations, and also supports Progressive Web App (PWA) implementation.

4.1.1 Hypertext Markup Language (HTML)

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

4.1.2 Cascade Style Sheets (CSS)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.[3] This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

For the project I have used SCSS (Sassy CSS) and used throughout this reference, is an extension of the syntax of CSS. SCSS is an advanced type of CSS which enables CSS pre-processing in runtime.

4.1.3 Javascript

JavaScript is a high-level, dynamic, un-typed, and interpreted programming language. It has been standardized in the ECMAScript language specification. Alongside HTML and CSS,

JavaScript is one of the three core technologies of World Wide Web content production; the majority of websites employ it, and all modern Web browsers support it without the need for plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

It has an API for working with text, arrays, dates and regular expressions, but does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded. Although there are strong outward similarities between JavaScript and Java, including language name, syntax, and

respective standard libraries, the two are distinct languages and differ greatly in their design. JavaScript was influenced by programming languages such as self and Scheme.

4.1.4 Typescript

TypeScript is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript and adds optional static typing to the language. TypeScript is designed for development of large applications and transcompiles to JavaScript. As TypeScript is a superset of JavaScript, existing JavaScript programs are also valid TypeScript programs.

TypeScript may be used to develop JavaScript applications for both client-side and server-side execution (as with Node.js or Deno). There are multiple options available for transcompilation. Either the default TypeScript Checker can be used, or the Babel compiler can be invoked to convert TypeScript to JavaScript.

TypeScript supports definition files that can contain type information of existing JavaScript libraries, much like C++ header files can describe the structure of existing object files. This enables other programs to use the values defined in the files as if they were statically typed TypeScript entities. There are third-party header files for popular libraries such as jQuery, MongoDB, and D3.js. TypeScript headers for the Node.js basic modules are also available, allowing development of Node.js programs within TypeScript.

4.1.5 AJAX

Ajax (also **AJAX** ; short for "Asynchronous JavaScript and XML") is a set of web development techniques using many web technologies on the client side to create asynchronous web applications. With Ajax, web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behaviour of the existing page. By decoupling the data interchange layer from the presentation layer, Ajax allows web pages and, by extension, web applications, to change content dynamically without the need to reload the entire page.[3] In practice, modern implementations commonly utilize JSON instead of XML.

Ajax is not a single technology, but rather a group of technologies. HTML and CSS can be used in combination to mark up and style information. The webpage can then be modified by JavaScript to dynamically display—and allow the user to interact with—the new information. The built-in XMLHttpRequest object, or since 2017 the new "fetch()" function within JavaScript, is commonly used to execute Ajax on webpages, allowing websites to load content onto the screen without refreshing the page. Ajax is not a new technology, or different language, just existing technologies used in new ways. In my project AJAX is used by Angular frontend to HTTP request handling with my RESTful API.

4.2 Development Tools and Technologies

4.2.1 Angular

Angular is an open source front end web framework which is based on JavaScript. This was maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications. It was built with the Model-View-Controller concept in mind. This framework introduced many features which allows the developer to create rich, single-page applications easily. The concept of data binding was also introduced which enables the automatic updates of the view whenever the model is changed.

4.2.2 Express JS

Express.js, or simply **Express**, is a web application framework for Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs.

It has been called the de facto standard server framework for Node.js.Express is the back-end component of the MEAN stack, together with the MongoDB database software and AngularJS front-end framework.

4.2.3 Node JS

Node.js is an open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser. Node.js allows developers to use JavaScript to produce dynamic web page content before the page is sent to the user's web browser. Node.js enables to use a unifying web application development around a single programming language, rather than different languages for server- and client-side scripts. Node.js is built on Chrome's V8 JavaScript engine. It provides an event driven, non-blocking (asynchronous) I/O and cross-platform runtime environment for building highly scalable server-side application using JavaScript.

4.2.4 MongoDB

MongoDB is a cross-platform and open-source document-oriented database, a kind of NoSQL database. As a NoSQL database, MongoDB shuns the relational database’s table-based structure to adapt JSON-like documents that have dynamic schemas which it calls BSON.

This makes data integration for certain types of applications faster and easier. MongoDB is built for scalability, high availability and performance from a single server deployment to large and complex multi-site infrastructures.

4.2.5 Mongoose

Mongoose is an Object Data Modeling (ODM) library for MongoDB and Node.js. It manages relationships between data, provides schema validation, and is used to translate between objects in code and the representation of those objects in MongoDB.MongoDB is a schema-less NoSQL document database. It means you can store JSON documents in it, and the structure of these documents can vary as it is not enforced like SQL databases. This is one of the advantages of using NoSQL as it speeds up application development and reduces the complexity of deployments.

4.2.6 Visual Studio Code IDE

Visual Studio Code (VSCode) is a source code editor developed by Microsoft that can be run on Windows, macOS, and Linux. It is free, open-source, and provides support for debugging as well as built-in Git version control, syntax highlights, snippets, and so on. The UI of VSCode is highly customizable, as users can switch to different themes, keyboard shortcuts, and preferences. VSCode was originally announced in 2015 as an open-source project hosted on GitHub before releasing to the web a year later. Since then, Microsoft's code editor has been gaining popularity among developers. In the Stack Overflow 2018 Developer Survey, VSCode was ranked as the most popular development environment with around 35% out of over 100,000 respondents saying they use it. More stunningly, this figure is around 39% in the web development field. And with monthly updates, users can expect to enjoy an even better experience—bug fixes, stability, and performance boosts are frequently pushed.

4.3 Third party Components/ Libraries

4.3.1 JWT (JSON Web Token)

JSON Web Token is an Internet standard for creating JSON-based access tokens that assert some number of claims. The tokens are signed by one party's private key, so that both parties are able to verify that the token is legitimate. The tokens are designed to be compact, URLsafe, and usable especially in a web-browser single-sign-on (SSO) context. JWT claims can be typically used to pass identity of authenticated users

between an identity provider and a service provider, or any other type of claims as required by business processes.

4.3.2 Socket.io

Socket.IO is a JavaScript library for realtime web applications. It enables realtime, bi-directional communication between web clients and servers. It has two parts: a client-side library that runs in the browser, and a server-side library for Node.js. Both components have a nearly identical API. Like Node.js, it is event-driven.

Socket.IO primarily uses the WebSocket protocol with polling as a fallback option,[3] while providing the same interface. Although it can be used as simply a wrapper for WebSocket, it provides many more features, including broadcasting to multiple sockets, storing data associated with each client, and asynchronous I/O.

4.3.3 Multer

Multer is a node.js middleware for handling multipart/form-data, which is primarily used for uploading files. It is written on top of busboy for maximum efficiency. Multer adds a body object and a file or files object to the request object. The body object contains the values of the text fields of the form, the file or files object contains the files uploaded via the form.

4.3.4 Nodemailer

Nodemailer is a module for Node.js applications to allow easy as cake email sending. The project got started back in 2010 when there was no sane option to send email messages, today it is the solution most Node.js users turn to by default.

4.3.5 Angular Material: UI elements

Angular Material is a UI component library for Angular JS developers. Angular Material components help in constructing attractive, consistent, and functional web pages and web applications while adhering to modern web design principles like browser portability, device independence, and graceful degradation. It helps in creating faster, beautiful, and responsive websites. It is inspired by the Google Material Design.

4.3.6 MDB UI component library

The MDB tools project is an open source effort to create a set of software libraries and utilities to manipulate files in the proprietary JET 3, 4 and 5 database formats (used by Microsoft Access.).

The MDB tools include Gnome MDB Viewer (gmdb2), a desktop tool for viewing MDB databases on Linux. Kexi (visual database applications creator) uses MDB tools for importing MDB databases. For Mac OS X there are three proprietary applications named MDB Explorer, MDB Viewer and MDBLite that also provides a graphical front end to the MDB tools library.

4.3.7 Bodyparser

Body-parser extract the entire body portion of an incoming request stream and exposes it on req. body. The middleware was a part of Express. js earlier but need to install it separately. This body-parser module parses the JSON, buffer, string and URL encoded data submitted using HTTP POST request.

4.3.8 Ngx-qrcode

This angular helper package is used to generate QR Codes which is required in my event invitation handling functionalities.

4.3.9 rxjs

Reactive programming is an asynchronous programming paradigm concerned with data streams and the propagation of change (Wikipedia). RxJS (Reactive Extensions for JavaScript) is a library for reactive programming using observables that makes it easier to compose asynchronous or callback-based code. See (RxJS Docs).
RxJS provides an implementation of the Observable type, which is needed until the type becomes part of the language and until browsers support it. The library also provides utility functions for creating and working with observables. These utility functions can be used for: Converting existing code for async operations into observables, Iterating through the values in a stream, Mapping values to different types, Filtering streams and most specially design and implementing Observer design pattern in my project.

4.3.10 Ngx-cron

This cron library was used to implement scheduled tasks, like monthly subscription fee generation for service providers, timely backup etc..

4.3.11 nodemon

This is a development – only library which is used to track and debug backend functionalities.

4.4 API services and other services

4.4.1 Google Maps API

A. Geo coder API
   This API was used to map service providers' and event locations on dashboards, location details retrieval for different purposes in the system.

B. Places API

This API was used identify places (ex:- cities, towns) with the reverse geocoding.

4.4.1 Paypal Sandbox

Paypal sandbox environment was integrated as the payment gateway for the system to handle online transactions which are processed through the system.

4.4.2 Mongodb Atlas

MongoDB Atlas is the realtime clustered database used for my system. This API provides a set of APIs' like MongoDB Charts for data analysis and visualization, mongoDB cloud CLI etc..

4.5 Design Principles and Patterns

Design patterns are commonly defined as time-tested solutions to recurring design problems. The term refers to both the description of a solution that you can read, and an instance of that solution as used to solve a particular problem.

4.5.1 Singleton Pattern

In software engineering, the singleton pattern is a software design pattern that restricts the instantiation of a class to one "single" instance. This is useful when exactly one object is needed to coordinate actions across the system. In the developed system, Angular service classes and objects initiated from these classes are following singleton pattern. A singleton service is a service for which only one instance exists in an app. There are two ways to make a service a singleton in Angular:

- Set the providedIn property of the @Injectable() to "root".
- Include the service in the AppModule or in a module that is only imported by the AppModule

4.5.2 Observer Design Pattern

Observer pattern uses three actor classes. Subject, Observer and Client. Subject is an object having methods to attach and detach observers to a client object. We have created an abstract class Observer and a concrete class Subject that is extending class Observer.

To avoid the situation that the application stop running to wait for data, I have put observable in between frontend and server. Advantage of this observable is script can request to observable for getting data from server and then continue with script no need to wait for data. Then observable get data from server and that data will pass to script. In specific cases I have used explicitly implemented observer patterns, but in most cases I have used Subject, Subscriber, Observable implementation which is provided by “rxjs” library.

4.6 Summary

In this chapter, the development aspects of the system has been elaborated under the areas of technologies, constraints, third party support and the maintenance of confidentiality. The factors such as availability, cost, performance, functionality, etc. have been considered in order to decide the technologies described above

CHAPTER 5

5 SYSTEM TESTING

Software testing aspects of the 'Evenza' event management and resourcing platform is discussed under this chapter. The strategies, types of testing used and why and how they have been employed to test the system have been discussed throughout this chapter. Furthermore, it discusses about the sample test cases, followed by the test reports of the system and results illustrating the severity of the bugs identified and possible solutions.

5.1 Test Plan and Test strategy

The functionalities are tested in order to advance the functions and the quality of the product. Testing is done with the purpose of aligning the product with the users' requirements. The testing plan defines the items to be tested and the functions which are selected based on the importance of the functions, and the risk of the functions on the user's viewpoint. Then the test cases were designed corresponding to the use case descriptions. They were executed manually, and the results were recorded. The bugs identified were corrected and tested again, where the functionalities were improved in "Evenza". In the testing process of this system both black box testing and white box testing has used.

5.1.1 Testing Approach

White box testing is mostly carried out while the coding is taking place and each statement, algorithm, condition and decision structures tested soon after the finishing of the particular code snippet. White box testing allows to identify the small bugs that could be emerged by the simple mistakes of the code and the errors in the decision logic. As well as it enables developers to recover from errors before integrating the system. Because of these reasons choosing white box testing for the initial coding phases was very expedient.

Black box testing is used once after a particular functionality is completely developed. Errors and bugs are identified by entering set of random values to the functionality and the bugs and errors identified are fixed and put into black box testing again and again until all bugs are covered. The gessoes over user errors can be tested at this stage. Under these testing approaches four main test types has used to test this system.

Concept testing – concept testing is used to check the compatibility of the concepts used in the system and the concepts in the actual business process.

Unit testing – This type of testing has used white box testing approach to test each and every critical units (i.e. functions, algorithms, etc...) built in the system individually. Only if they are passed these tests, units are integrated with each other and tested in integration testing.

Integration testing - After integrating the components together which are tested using the unit testing the entire integrated component is tested again. Both black box and white box testing approaches have used for this testing type.

Interface testing – Interface testing has used to test each and every component in the GUIs against their intended purpose.

5.2 Sample Test Cases

Table 5.1 Test Case: User Registration

1. User Registration (Basic sign up)				
ID	Description	Input	Expected Output	Result
1.1	Regular user Registration Entering details of a new user who have not registered before in the system. (Customer, Merchant)	Firstname, Lastname, Email,Password	If Input data is valid, Customer is signed up and, signup successful message displayed. Merchant is directed to the next step registration page. Entered information passed to the next step registration page	Pass
1.2	Validating user registration: Entering pre-registered email in the sign up	Firstname, Lastname, Existing email, password	Show in red color under email field that this email has already taken	Pass
1.3	Validating user registration: Entering the password less than 8 characters, not including numbers in the signup screen	Firstname, Lastname, Email, Password less than 8 characters, no numeric characters provided	Show error in red color under the password field	Pass
1.4	Validating user registration: Entering an unrecognized email address	Firstname, Lastname, Email (Not in the desired format)	Show error in red color under the email field	Pass

Table 5.2 Test Case: User Login

2. User Login (Customer, Merchant, Admin)				
ID	Description	Input	Expected Output	Result
1.1	Regular user Login Entering user name and password (Customer, Event planner, Merchant)	Existing Email, Password	If Input data is valid, user is signed in by generating a secure token for authentication using JWT, and redirected to the related dashboard.	Pass
1.2	Validating user registration: Entering email in an unrecognized format	Existing email (invalid format), password	Show in red color under email field that this email is invalid	Pass
1.3	Validating user registration: Entering an invalid username or password	Existing Email, Password	Show error message saying Invalid username or Password	Pass

Table 5.3 Test Case: Creating a Business Profile

3. Create a Business Profile (Merchant)				
ID	Description	Input	Expected Output	Result
1.1	Creating a business profile: Input all required details in valid format	Business Title, Logo (Image), Cover Photo (Image)Description, Address, Contact No, Business Email, Location (Selected on Map), Open Days (Open times in a week)	If no business profile is created for the user, a new business profile is created with input data, success message is shown, redirected to the newly created profile	Pass
1.2	Validating user inputs: Entered data in invalid format	Business Title, Logo (Image), Cover Photo (Image)Description, Address, Contact No, Business Email, Location (Selected on Map), Open Days (Open times in a week)	Show information in red color under relevant field that this input data is invalid. No form submission until all the inputs are valid	Pass
1.3	Validating Service Creation: Images are not uploaded	All input data except Logo or Cover photo.	Create business profile with assigning default images for placeholders where images are not provided	Pass

Table 5.4 Test Case: Adding a new Service

4. Add a new service (Merchant: Service Provider)				
ID	Description	Input	Expected Output	Result
1.1	Create a new Service: Input all required details in valid format	Service name, Rate, Rate type (Per hour, per day), capacity, availability for bookings, availability for appointments, Feature Images, Description, Category	Service Id checked and generated, new service is created and appeared in related places.	Pass
1.2	Validating user inputs: Entered data in invalid format	Service name, Rate, Rate type (Per hour, per day), capacity, availability for bookings, availability for appointments, Feature Images, Description, Category	Show information in red color under relevant field that this input data is invalid. No form submission until all the inputs are valid	Pass
1.3	Validating Service Creation: Feature images are not uploaded	Service name, Rate, Rate type (Per hour, per day), capacity, availability for bookings, availability for appointments, Description, Category	Create service with assigning default images for placeholders where images are not provided	Pass

Table 5.5 Test Case: Add a new Product

5. Add a new product (Merchant: Product Seller)				
ID	Description	Input	Expected Output	Result
1.1	Create a new Product: Input all required details in valid format	Product name, Rate, Quantity type, Price, inventory, availability for ordering, Feature Images, Description, Category	Product Id checked and generated, new product is created and appeared in related places.	Pass
1.2	Validating user inputs: Entered data in invalid format	Product name, Rate, Quantity type, Price, inventory, availability for ordering, Feature Images, Description, Category	Show information in red color under relevant field that this input data is invalid. No form submission until all the inputs are valid	Pass
1.3	Validating Product Creation: Feature images are not uploaded	Product name, Rate, Quantity type, Price, inventory, availability for ordering, Description, Category	Create product with assigning default images for placeholders where images are not provided	Pass

Table 5.6 Test Case: Create a new booking

6. Create New Booking (Customer)				
ID	Description	Input	Expected Output	Result
1.1	Create a new Booking: Input all required details in valid format	From Date & Time, To Date & Time, (time in No of Hours per day, 10 hrs+ considered as a full day) Service to book, Paying Amount, Additional comment message	Compare both dates, and the capacity for bookings for given inputs, if valid redirect to payment page	Pass
1.2	Validating booking creation :Input data is invalid	From Date & Time, To Date & Time, Service to book, Paying Amount, Additional comment message	Show in red color under related field that input data is invalid	Pass
1.3	Validating booking creation :booking availability on selected dates, capacity	From Date & Time, To Date & Time, Service to book, Paying Amount, Additional comment message	Run checkAvailability() function, show a error message if service is not available for booking.	Pass
1.4	Validating booking creation :booking Payment is invalid (not paid through the card)	From Date & Time, To Date & Time, Service to book, Paying Amount, Additional comment message	Show error is red color if paying amount is less than 10% of total amount. Halt database transaction for creating the booking, if the payment is not paid through the payment gateway.	Pass

1.5	Validating booking creation :booking Payment is valid through payment gateway	Paying Amount, Payment Method (Credit Card/ Paypal)	Update payment details and generate a booking receipt	Pass

Table 5.7 Test Case: Create a new Visit Appointment

7. Create New Appointment (Customer)				
ID	Description	Input	Expected Output	Result
1.1	Create a new Appointment: Input all required details in valid format	Appointment Date & Time, Service to make a visit appointment, Additional comment message	Compare both dates, and the capacity for bookings for given inputs, if valid redirect to Appointment note page	Pass
1.2	Validating appointment creation :Input data is invalid	Appointment Date & Time, Service to make a visit appointment, Additional comment message	Show in red color under related field that input data is invalid	Pass

Table 5.8 Test Case: Create a new Product Order

8. Create New Product Order (Customer)				
ID	Description	Input	Expected Output	Result
1.1	Create a new Product Order: Input all required details in valid format	Product, Quantity, Delivery Address, Paying Amount, Additional comment message	Compare with inventory and if available redirect to payment page	Pass
1.2	Validating product order creation: Input data is invalid	Product, Quantity, Delivery Address, Paying Amount, Additional comment message	Show in red color under related field that input data is invalid	Pass
1.3	Validating product order creation: Inventory availability	Product, Quantity, Delivery Address, Paying Amount, Additional comment message	Inventory is displayed, if quantity input box is not allowed enter value greater than inventory	Pass
1.4	Validating product order creation: Payment is invalid (not paid through the card)	Product, Quantity, Delivery Address, Paying Amount, Additional comment message	Show error is red color if paying amount is less than 10% of total amount. Halt database transaction for creating the order, if the payment is not paid through the payment gateway.	Pass

Table 5.9 Creating a new event

1. Create New Event (Customer)				
ID	Description	Input	Expected Output	Result
1.1	Create a new Event: Input all required details in valid format	From Date & Time, To Date & Time, no of participants, total budget, select required service and product categories, social media links, description, location, feature image	Compare both dates, and the capacity for bookings for given inputs, if valid redirect to payment page	Pass
1.2	Validating event creation :Input data is invalid	From Date & Time, To Date & Time, no of participants, total budget, select required service and product categories, social media links, description, location, feature image	Show in red color under related field that input data is invalid	Pass
1.3	Validating booking creation :booking only a few/ no suggested service/ product categories selected	Service, product categories	Align budget allocations accordingly to selected services,	Pass

1. Add Event Guests (Customer)				
ID	Description	Input	Expected Output	Result
1.1	Add a new Guest: Input all required details in valid format	Name, Email	Add the guest to guest list, make event state to unpublished, schedule invitation and other emails to be sent to the guest, when the event is publishing	Pass
1.2	Validating guest adding :Input data is invalid	Name, Email	Show in red color under related field that input data is invalid	Pass

5.3 Test Report

Testing is done as part of the development process, and not at the end of the whole process. If it is not done throughout the development process then the testing is done at the end, if so the system will take many changes even in the interface design and process of interface navigation too. Thus changing these at the end is costly and less effective. The number of defects that is uncovered by the test cases and test data is decrease gradually with the number of testing iterations performed. In the initial iterations large number of bugs uncovered and most of those errors are moderate interims of importance. Bugs which were found during the tests were fixed immediately. If it could not be fixed then and there, it was written down and retried later. Since the tests were done in an iterative manner, bugs were reduced along the time.

Testing Methods used: Debugging and Logging were used to check API communication and data send/ retrievals were correctly happening. I have also used Postman as the API Testing tool.

5.3.1 Test Environment

Hardware:

- Processor: Intel(R) Core(TM) i5-7030U CPU @ 2.10GHz, 2200 Mhz,
- RAM: 8GB
- SSD : 128 SSD

Software:

- OS: Debian Ubuntu 18.04
- Browser: Google Chrome-Version 69.0.3497.100 (Official Build) (64-bit)

Debugger and testing tools

- Google chrome development tools, inspect element facility/ browser console
- Karma, jasmine (Angular frontend unit testing)
- TsLint, EsLint, Angular, Node and JS Snippet extensions for IDE
- Debugging and login using IDE
- Postman for API testing

5.4 Summary

In this chapter we discussed about the types of tests which are the developer has followed when implementing the testing procedure and test cases of the system. Furthermore, the test strategies used, test plan, sample test cases and the report of the overall test phase are described.

CHAPTER 6

6 SYSTEM IMPLEMENTATION

This chapter consists of the implementation details of “Evenza”, smart event management and Resourcing Platform. It will discuss about the implementation specifications of the system such as software requirements and hardware requirements. As well as it will further discuss about installation guide and security procedures. This also includes a User Manual that guides the user to how to use the system.

6.1 System Requirements

This section consists of the Hardware and Software requirements of the proposed system.

6.1.1. Hardware Requirements

Minimum hardware requirements of the client machine (To run the web Browser)

- CPU - Pentium Dual Core above 2.8 GHz
- 1 GB or above RAM
- Router with firewall to be always connected to the internet
- 128 MB VGA Card
- Mobile PWA – 5MB Phone Storage

Web service requirements

- Connected to Internet

6.1.2 Software Requirements

Operating system – Windows 7+, Linux distro, or Mac PC,

Mobile PWA – Android, iOS (any compatible version)

For System Administrator (to host the System)

Access to Paas Cloud Web Service (Google, Azure Amazon or related)

Or Cloud virtual Machine / On Premise Server

- 400 MB Storage Space (Scalable according to user data)
- Core i3 Processor
- 2 GB RAM
- Angular (7.* +) , Node(10.* +) and npm package manager

6.2 Installation guide

For a general user, the system can be accessed via Internet, navigating to **http://www.evenza.biz**

For an Administrator to host the project, or to build the project locally, follow these instructions.

1.Download the master branch of project repository, if access is given.

2. Extract the project file to a new directory.

3. Open a terminal/ Command Prompt, navigate to the directory, and run **ng build** to build the project

4. Run **npm run** to run both frontend and backend same time or run **npm run** serve to run the backend separately.

5. Go to localhost:4200 to access the system.

(No need to configure the database, as database hosted in cloud)

6.3 User Access Levels

There are 3 levels of users

6.3.1 Admin

Access Level Rights –

- Add/ Delete/ Edit/ manage the users.
- Collect payments, handle subscriptions
- Add/ Remove/ Modify Service, Event & Product categories/ plans through the system

6.3.2 Merchants (Service Providers/ Sellers)

Access Level Rights –

- manage their user profiles
- Add/ Remove/ edit services/ products
- Approve/ Cancel lord Mark as Delivered on Service Bookings, Appointments & Product orders

6.3.3 Customer

Access Level Rights –

- manage their user profiles
- Add/ Remove/ edit events
- Request cancel orders for pending appointments, bookings
- Add Guests to the event and invite them
- Schedule / edit / remove event tasks

6.4 User Manual

This user manual will guide you through the system functionality enabling you easily manage the 'Evenza', smart online event management and resourcing platform.

6.4.1 Customer (Event Planner) User Manual

Step 1: To get started, visit http://evenza.biz

Evenza Homepage will appear as below

Figure 6.1 Evenza homepage

Step 2: Click on **Sign Up** to sign up as a user to the system.

Figure 6.2 Sign up type select screen

Step 3: Fill in the required details and create your account.

Figure 6.3 User sign Up Details

Step 4: Once successfully signed up, Log in to System by Clicking on **Log In** on the top bar.

Figure 6.4 User Login Screen

Step 5: Your Dashboard will appear as below. On top of the dashboard, you can find quick links to access different services provided by the System. You can

- Create a new event or
- Directly search and book/ purchase Services and Products

And also you can access your **Booking/ Order and Service Appointment history**, **User Profile and Terms & conditions, Help & Support** pages as well through the dashboard. At the bottom you can quickly access your latest **created events** as well.

Figure 6.5 Evenza user Dashboard

Step 6: You can also view your events in **My Events** tab in the side bar appears. To create a new event, click **on +Create New Event** button. Then select an Event Type.

Figure 6.6 Select Event Category

Step 7: Fill the required details and select required Service and Product categories to be included in the event plan. You can also set event location, planned budget, number of expected participants erc.. through following steps

Figure 6.7 Create an Event - step 1

Figure 6.8 Create an Event - Step 2

Figure 6.9 Create an Event - Select Service and Product Categories required

Figure 6.10 Create an event - Summary view before creation

Step 8: Once an Event is Created, you can view event details, share it through social media, and manage event in **Event Details** view as below.

Once the event is published details can be viewed publicly by visiting

http://evenza.biz/events/event_id

- **Edit Event Details-** You can edit event details like dates, no of participants, budgets accordingly. Changes will effect to whole event. If the event is **published**, if modified, it will be automatically set to **Unpubliahed**. You can **Publish the Event** , once all the modifications are done.
- **View event Plan** – view and manage reserd service bookings, product purchases and scheduled tasks in the Event Plan.
- **View event Schedule** – View and Manage event related tasks, bookings and appointments in a calendar view, get upcoming event tasks notifications, schedule and manage event related tasks.
- **Manage Participants** – Add event participants (Closed type events) or Create and share Open type events registration form, create and share event Invitation, QR access codes and view/ print event participants.

Figure 6.11 event Details View

Event Plan – You can schedule your own tasks, order them by dragging and dropping, Allocated service and product categories will be listed. You can select and book/ purchase your preferable service or product through this plan. Budget and other details will be automatically adjusted and managed according to your reservations.

Figure 6.12 Event Plan: Task List

Figure 6.13 Create New task dialog Box

Figure 6.7 Event Plan : Service and Product Bookings, purchases & allocations with budget

Event Schedule – Schedule and manage tasks, view bookings, appointments and tasks on calendar and also this space is used to show scheduled task alerts (Within a day, 6 hours and 3 hours). Alerts will also be notified via emails.

Figure 6.15 Event Schedule

Participants Management Area – Here you can create and update event invitation, which will be shared among all the guests, and also add and manage event guests. For open type events, you can create and manage Event Registration Form, and also those registered participants details can be seen here. Participant list can be printed or retrieved via email as a soft copy.

Figure 6.16 Participant Management: Invitation

Figure 6.17 Adding Guests and the guest List View

Publish an Event: Once an event is modified, or new guests are added, Event will be set to Unpublished state. After all the modification you can Publish the event.

When publishing an event, the system will

- Send Invitations to all the guests (If invitation has not changed, only for newly added guests)
- Update all event details and plans accordingly.
- Modified dates will be notified to service providers, if services are booked on those dates.

Figure 6.18 Event Publish Dialog

Step 9: You can place and order, booking or a visit appointment by selecting the desired product or service. If any promotions are stated, promotions and coupons will be automatically deducted from total fee. Payment can be made with either credit/ debit card or PayPal.

Figure 6.19 Service Search

Figure 6.20 Product Search

Step 10: You can view and manage your booking, visit appointments and product orders through the dashboard, If an order or booking in pending state, you can request for a cancellation. Order will be cancelled only if it's accepted by Service Provider, Paid amount will be refunded.

Figure 6.21 Manage Event bookings, Appointments and Orders

Step 11: You can view Real Time Event Management Dashboard, important analytical and comparative data visualizations by visiting **Event Report Dashboard**. Relevant reports can also be generated.

Figure 6.22 Event Report Dashboard

Step 12: A created event can be cancelled. In cancellation process, If the event has been published at least once,

- All the invited participants will be informed with a cancellation notice via an email.
- Pending services state will be updated and relevant service providers will be notified.
- This action can be performed only once and cannot be rolled back.

Figure 6.23 Cancel Event Dialog

6.4.2 Merchant (Service Providers and Sellers) User Manual

Step 1: To get started, visit http://evenza.biz

Evenza Homepage will appear as below

Figure 6.25 Evenza homepage

Step 2: Click on **Sign Up** to sign up as a Merchant to the system.

Figure 6.26 Sign up type select screen

Step 3: Fill in the required details and create your account.

Figure 6.27 Sign up: Provider user details

Step 4: Once successfully signed up, Log in to System by Clicking on **Log In** on the top bar.

Figure 6.28 User Login Screen

Step 5: Once logged in, your dashboard will appear as below. In the Home page it will show Realtime sales orders & appointments related statistics and a list of recent bookings or orders.

Figure 6.29 Service Provider Dashboard

Step 6: In order to merchandize through the platform, you need to **create a business profile** first, First you need to upload copies of your NIC. After reviewing them by System Admins, you will be given chance to create a business profile. One Service Provider can have only one business profile. After creation, you can mark your business as **Verified (Trusted)** by uploading your BR copy and other certification for a review.

Figure 6.30 ID Verification required before creating a business profile

Figure 6.31 Submit NIC Copies for verification

Figure 6.32 Create the Business Profile

Figure 6.33 Profile view after creation

Step 7: You can set Business Opened Days, Location, Credit/Debit card details or PayPal account details while creating your business profile.

Figure 0.2 Setting Business Open Days

Figure 6.34 Set Business Opened Days

Figure 6.35 Setting Business Location

Step 8: Then you can add services or products (according to your user type) under the business profile, Modify, remove or Add promotions.

Figure 6.36 Add new Service

Figure 6.37 Adding Promotion to a Product or service

Step 9: You can view received Service order details, bookings and visit appointment details on the dashboard. You can also make any pending order or a booking to either Completed/ Delivered state if delivered or completed, or else Cancel them. If they are cancelled, Payment should be refunded.

Figure 6.38 View List of Bookings

Figure 6.39 Cancel an order

Step 10: Booking Calendar is a special feature provided for Service Providers. If they have booking on their services provided through the platform, by a third party, then they can reserve those dates, time slots using the booking calendar. If so, those dates and times for the relevant service will not be available for booking.

Figure 6.40 Booking Calendar

Step 11: All Service providers, sellers are required to pay a monthly subscription fee since the first month of their registration. First month will be subscription free – as a trial month. Payment section can be found right next to your business profile. If the payments are overdue, You may loose your Sales profits and Account access.

Figure 6.41 Payment section

Figure 6.42 Pay subscription Fee

Step 12: Sellers can manage their **Product Inventory and Assigned Delivery Service Details.** Delivery charges are charged by customers and debited to sellers' account when sale is happening. Payments made to Delivery Services and other agreements are explicit form the system.

Figure 6.43 Handle Delivery Services and Product Inventory

Step 13: Service Providers and Sellers can generate different Business Reports according to their requirements.

Figure 6.44 Generating Business Reports

6.4.3 System Administrator (Owner) User Manual

Step 1: System Administrator accounts are created once and done by Developers, There is no sign up for System Admins. Expected to have only one administrator. But depend on the requirements, system can have many. All user level access and privileges are Provided in System Administrator Account.

Figure 6.45 Admin can also sign in as usual with their credentials

Step 2: After Sign in Admin is capable of,

Figure 6.45 Admin can also sign in as usual with their credentials

- View Overall business transactions, and Summary of data as Visualizations Realtime.
- View Payment collections and Payment history of different Service Providers.
- Warn or remove Service Providers on violation of terms or not paying due subscription fee.
- Add/ edit Service and Product Categories. Event Categories (Allocating relevant Service and Product categories for an event with initial budget percentages)
- Create Database backups and restore on demand.
- Review and Approve Identity/ Business Verifications.
- Generate and view System wide Business reports and Dashboards.
- Manage their user profiles as usual as Service Providers/ Customers.

Below figure will show System Administrator's capabilities as described above.

Figure 6.46 View Overall Transaction Visualizations

Figure 6.47 View and Manage Subscription Payments

Figure 6.48 view and Manage Service Providers/ Sellers

Figure 6.49 Create and manage Service/ Product and Event Categories

Figure 4.50 Create Backups and restore

Figure 6.51 Verify ID and Business Verification requests

Figure 6.52 View and manage overall dashboards and business reports

CHAPTER 7

7 EVALUATION AND CONCLUSION

This chapter evaluates the attempt taken in the project to deliver the system analysis and designing approaches and provides a conclusion of the project. This includes degree of objectives met within the development of the system, limitations and drawbacks, and future modifications planned to be made, improvements and extensions possible.

7.1 Degree of Objectives Met

Prior to the beginning of the project, there were few objectives that were set for achievement. Following section is about to which extent they are achieved and the justification for it.

1. Give service providers and suppliers a platform to advertise their goods and services online.

    Through this system service providers and sellers are able to manage their business online, add their services and products, add promotions on demand and their services and products are accessible to their target market, customers who are willing to plan or organize an event.

2. Enable event planning and organizing to do so easily, effectively and efficiently.

    Through the system users can efficiently create and manage events, invite participants, share through social media and and manage event budget accordingly to the services hire or purchase.

3. Enable event planners to monitor what they have ordered online and tackle their event plan.

This objective also met as customers (event planners) can view and manage their event plan, set the event state to published, unpublished, create and get notified on scheduled tasks.

7.2 Cross Checking the Objectives

7.2.1 Functional Requirements

Table 7.1 Cross Checking objectives met with Functional Requirements

ID	Requirement	
1	Shall be able to create and manage user accounts.	Done
2	Shall be able to login to the system.	Done
3	Shall be able to create business profiles (by service providers)	Done
4	Shall be able to manage products/services (by service providers)	Done
5	Shall be able to provide a set of event types to select.	Done
6	Shall be able to get event details for selected event type from customers.	Done
7	Shall be able to generate an event plan for selected event type.	Done
8	Shall be able to search for products/services under specific event plan segments.	Done
9	Shall be able to book/request appointments for services (by customers)	Done
10	Shall be able to order products (by customers)	Done
11	Shall be able to send cancellation/ modification requests for bookings/appointments (by customers)	Done
12	Shall be able to respond for bookings/appointments (by service providers)	Done
13	Shall be able to provide appointment/booking calendar for service providers.	Done
14	Shall be able to record booking and purchase history.	Done
15	Shall be able to show notifications related to event plan.	Done

16	Shall be able to record event plan details.	Done
17	Shall be able to get customer feedback (Rating and Reviews)	Done
18	Shall be able to generate reports.	Done
19	Shall be able to send invitations to event participants.	Done
20	Shall be able to send event related notifications to participants.	Done
21	Should be able to get participants registered to the open type events.	Done
22	Shall be able to backup data.	Done
23	Should be able to manage registered users.	Done
24	Should be able to receive payments (subscription fee and commission).	Done

With all above functionalities,

- The system improves efficiency and effectiveness of the event management and resourcing process through automation
- The system uses a centrally stored cloud-based database and therefore information available will be standard and consistent to everyone who has access to it.
- The system supports and enhances better and smother interaction among the users and it makes the communication more convenient.

Thus, it is justifiable to state that the system has met its objectives.

7.3 Usability, Accessibility, Reliability and Friendliness

From the commencement of the system design and the development process, usability aspects of the system has been considered, especially since the system is developed for a general users who may not tech-savvy. All the technical aspects of the system have been hidden from the system user through the interactive user interfaces. Developer has followed effective pre-defined principles when designing user interfaces in order to improve usability of the user interfaces of the Web Application and the Progressive Web Application (PWA) to aid the user to avoid errors through error and warning messages.

In addition, the system speaks the users' language and follows the natural and logical order of precedence according to the users' behavior. System also maintains the consistency throughout, use standard styles in all pages, to avoid complications.

Enabling and disabling options when necessary, use of drop down list to select from menus are some of error prevention methods used. Use of regular expression validation and scripts and reactive form validation, implementing interfaces has prevented users from entering invalid data. System provides a comprehensive user guide with step by step description with pictures. Error Messages and Validations are also described in the user guide so that the user can refer them if the error message is not informative enough.

In addition, the testing has been done throughout all the design and implementation phases. So it can be concluded that the user friendliness and the flexibility of 'Evenza', the smart event management and resourcing platform is satisfiable.

7.4 Limitations and Drawbacks

With the time and resources available "Evenza" has some limitations and drawbacks in its system with the current completion state of system development.

The event related service suppliers, event organizers, planners and small-scale service providers who owns SME (Small and medium scale) bushiness whom were met for requirement gathering, were lack of knowledge on IT solutions and they are more interested in manual process than computerizing. Many bookings are made manually on personal contacts and recommendations. The booking history is recorded manually even they could get aid of available booking and scheduling platforms. Therefore, it took considerable time and effort to gather all the actual business processes together and understand the interrelationships. However still there are some gray area to be discovered. So to overcome that, system has developed in and unstructured way, properly modeled, following design principles to enable instance extends upon future requirements.

Another drawback was the different platforms which are already used by different service providers for manage service bookings. Once the booking is made for the particular venue or the service on a specific date, the service providers are required to manually update their availability status in the calendars as it is not possible to communicate between different platforms they use, developed on different technologies. To overcome that, an interactive booking calendar feature has been introduced to the service providers, to reserve time slots , by their end.

Due to the technical capabilities required on machine learning and social media data mining techniques, system is unable provide a complete accurate and human like functioning chatbot service for event planning at first stage. Limitation of time for development also act as a barrier. Developing A community-based platform to connect generic parties in the system, takes more time and, more researches to be made for an accurate and universally implementable system.

7.5 Future Modification, Improvements and Extensions

There are several further modifications to the system design, which will enhance the quality of the system in terms of usability, reliability and accessibility.

- Integrate collaborative filtering, data mining techniques for recommendations.
- Integrate AI based business Intelligence, and forecasting tools using machine learning concepts.
- Provide an AI chatbot & voicebot service to quickly gather user requirements and provide a more customized event planning and resourcing experience.
- Introduce a community social network to the system for event publishing and sharing purposes.
- Provide more accessibility features with smart interactive interfaces.

7.6 Summary

In this chapter, the author has described about the degree of objectives met after implementing the project, limitations and drawbacks, and further development of the project.

REFERENCES

Creately (2019) *Use Case diagrams,* Available at: *https://creately.com/diagram-type/use-case* (Accessed: 20 November 2019).

Creately (2019) *Activity Diagram tutorial,* Available at: *https://creately.com/blog /diagrams/activity-diagram-tutorial/* (Accessed: 20 November 2019).

Visual Padigram Inc. (2019) *Visual Padigram - Information Systems Modeling Training,* Available at: *https://www.visual-paradigm.com/training/* (Accessed: September 2019).

Block Events (2019) Block Events - Blockchain based Online event Management Platform, Available at: https://blockevent.tk (Accessed: August 2019).

Eventray (2019) *Eventray - Event Management Platform,* Available at: *https://eventray.com/features* (Accessed: September 2019).

Angular Team, Google Inc. (2020) *Angular 8,9 - Documentation,* Available at: *https://angular.io/guide/displaying-data* (Accessed: February 2020).

Express & Node JS Maintainers (2020) *Express & Node JS Documentation,* Available at: *https://expressjs.com/en/starter/installing.html* (Accessed: March 2020).

mongoose Js (2020) *Mongoose JS Documentation,* Available at: *https://mongoosejs.com/docs/api.html* (Accessed: March 2020).

mongoDB University (2020) *MongoDB Documentation,* Available at: *https://docs.mongodb.com/* (Accessed: April 2020).