Start to Finish: A City Tour Application

SOFTWARE ENGINEERING PROJECT DOCUMENT

Gokul. S

Roll No: 2018103026

Batch: P

Srihari. S

Roll No: 2018103601

Batch: P

TABLE OF CONTENTS

SNO	TITLE	PAGE NO
1	Abstract	2
2	Scope	2
3	Objectives	3
4	Project Functionalities	3
5	SDLC Model Chosen: Spiral	4
6	Spiral Model: Context	7
7	Motivation to Spiral Model	9
8	Applications	10
9	Pros and Cons	11
10	Requirements Theory	12
11	Critical Non-Functional Requirements to our System	13
12	Iteration 1 (Inner Spiral)	14
13	Iteration 2(Middle Spiral)	24
14	Iteration 3 (Outer Spiral)	34
15	Current System and Limitations	40
16	Entity Relationship Diagram	41
17	Final Database Design	42
18	System Security	44
19	Conclusion	45
20	Future Enhancements	45
21	References	46

ABSTRACT:

Start to Finish is a dynamic application for an extremely important business domain: “tourism”. It is a dynamic and responsive mobile/web application that has been designed in a very effective manner to ensure the fact that every single aspect of a journey is put under a single roof. Nearly everyone goes on short leisure trips. For this, a City Tourism Management System would play a vital role in planning the perfect trip. The system allows its users to access all the required details such as location, events, etc. The main purpose here is to help tourism companies to manage customers, transport, accommodation, food, etc. The system is to be used for single-day trips around the city. The proposed system maintains a centralized repository to make necessary travel arrangements and to retrieve information easily. The all-the-more cumbersome process of finding the suitable package and booking it has been tackled in an extremely careful manner so that communication between the tour operator and the user happens to be nothing less than a piece-of-cake.

SCOPE:

Start to Finish is an integrated software developed for tour operating companies. The main aim of this project is to help the tourism companies to manage their customers, vehicles, and agents. It makes all operations of the tour company easy and accurate. The standalone platform makes tourism management easy by handing agencies’ requests and providing servers for the customers located at different parts of various cities. Different modules have been incorporated in this project to handle different parts and sectors of the tour management field.

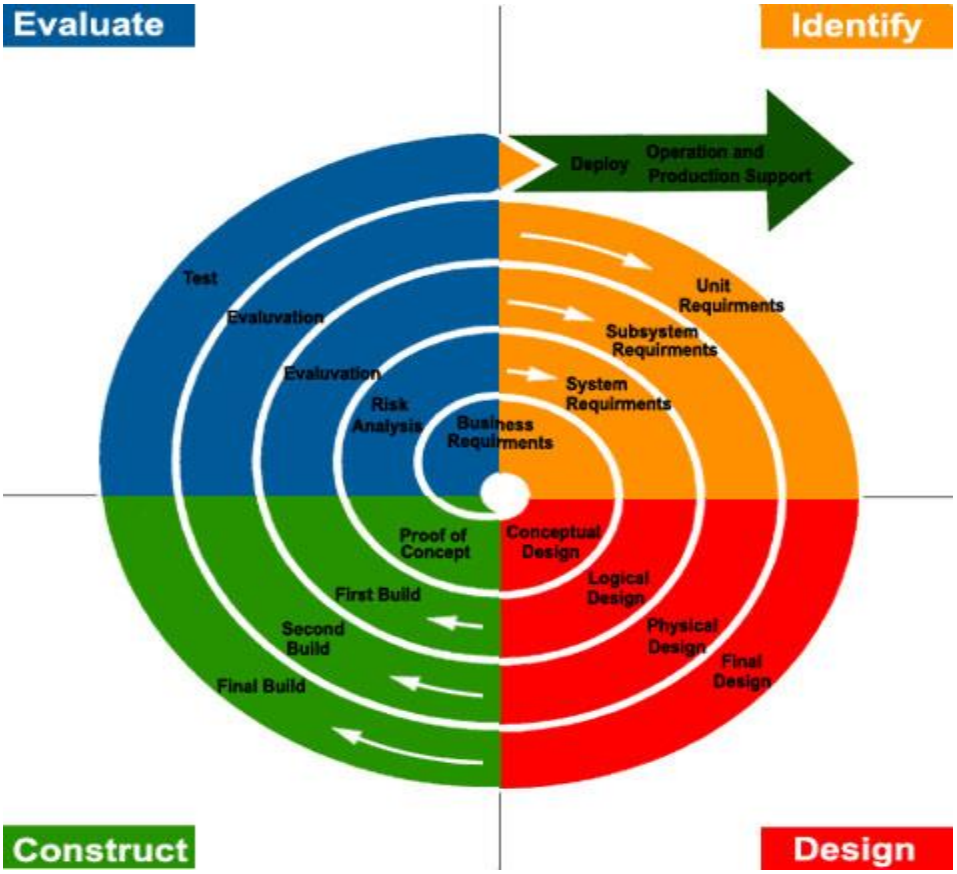
OBJECTIVES:

The proposed system is a mobile based application and maintains a centralized repository of all related details. The objective of this project is to develop a system that completely automates the processes and activities of a travel agency and customer details. The purpose is to design a system using which one can perform all operations related to travelling and sightseeing. With our incredible network all over the world, **Start to Finish** should match our clients with experts suited to your customer's interests and dynamics.

PROJECT FUNCTIONALITIES:

Feedback from tourists is taken in order to constantly improve the service. A plethora of inbound and outbound tour packages are made available to the customers to choose from. The details of the trip are recorded and maintained, which are used to suggest trips to the user based on their interests. Attractive tourist spots are made available for the passengers to choose from. Bonding different cultures gives the user an unique experience. This System also helps to develop tourism with different cultures so that they enrich the tourists' experience and build pride. A user interface is designed to enable users to plan their trip either based on the available packages. Provisions to book customized tours are provided to the elite customers. Finally, the ultimate aim of the system is to enrich the user's feel and make the workflow of the travel agency more smoother.

SOFTWARE DEVELOPMENT LIFECYCLE MODEL TYPE CHOSEN: SPIRAL



The spiral model is commonly known as an evolutionary development process. Commonly used as a lifecycle model for software development, the spiral model is similar to the iterative design process as there are repeated iterations (called cycles) in which successive attempts are made to develop a solution. However, the spiral model differs from the iterative model in a number of key areas. First, while the iterative research design process generally involves iterating on

prototype construction, the spiral model is more focused on risk reduction. Second, because the spiral model is based upon evolutionary software development, the spiral model expects and plans for a specific number of iterations whereas one of the limitations of the iterative model is that it is impossible to know when you have reached the best solution. With that said, the spiral model is also capable of supporting further iterations after the original planned iterations, similar to the iterative model. Finally, the key element of the spiral model is that it involves planning and executing different tasks during each iteration, on an as-needed basis as the project evolves. This is in sharp contrast to the use of the iterative model as a research design process, where the process simply involves repeated prototypes until the desired solution is built. As a research design process, the spiral model can be attractive because it makes you think about what steps should be taken first, what criteria needs to be met in order to move to a different phase (cycle) of the research process, what kinds of activities you would undertake if and/or when you meet those criteria, and what the risks are in doing these things. The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral.

The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals.

IDENTIFICATION

This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase.

This phase also includes understanding the system requirements by continuous communication between the customer and the system analyst. At the end of the spiral, the product is deployed in the identified market.

DESIGN

The Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and the final design in the subsequent spirals.

CONSTRUCT OR BUILD

The Construct phase refers to production of the actual software product at every spiral. In the baseline spiral, when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback. Then in the subsequent spirals with higher clarity on requirements and design details a working model of the software called build is produced with a version number. These builds are sent to the customer for feedback.

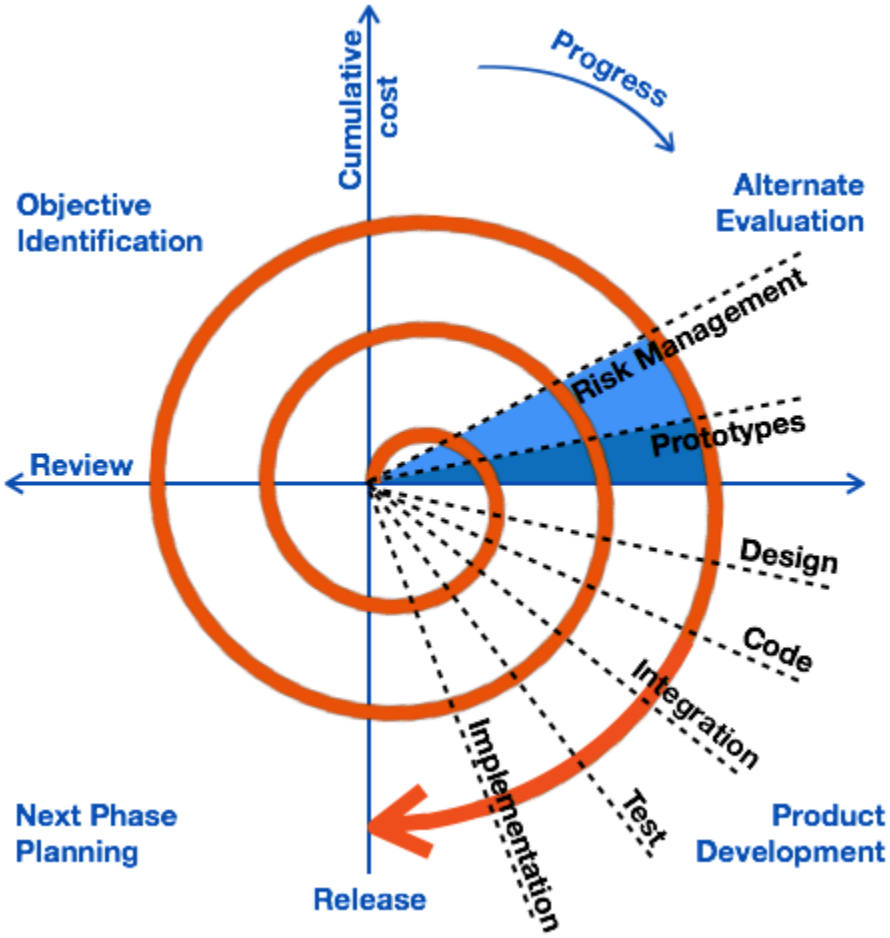
EVALUATION AND RISK ANALYSIS

Risk Analysis includes identifying, estimating and monitoring the technical feasibility and management risks, such as schedule slippage and cost overrun.

After testing the build, at the end of the first iteration, the customer evaluates the software and provides feedback.

Based on the customer evaluation, the software development process enters the next iteration and subsequently follows the linear approach to implement the feedback suggested by the customer. The process of iterations along the spiral continues throughout the life of the software.

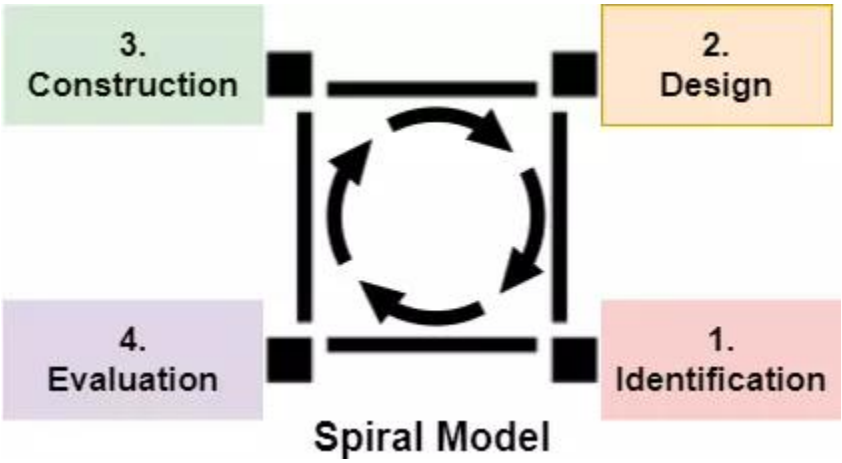
SPIRAL MODEL: CONTEXT



The spiral development model is a risk driven process model generator that is used to guide multi-stakeholder concurrent engineering of software-intensive systems. It has two main distinguishing features. One is a cyclic approach for incrementally growing a system's degree of definition and implementation while decreasing its degree of risk. The other is a set of anchor point milestones for ensuring stakeholder commitment to feasible and mutually satisfactory system solutions. The highlighted terms deserve further explanation: Risks are situations or possible events that can cause a project to fail to meet its goals. They range in impact from trivial to fatal and in likelihood from certain to improbable. Since risk considerations dictate the path a development must take, it is important that those risks be cataloged candidly and completely. See the references for taxonomy of risks and a method for identifying them . A process model answers two main questions: What should be done next? How long should it continue? Under the spiral model the answers to these questions are driven by risk considerations and vary from project to project and sometimes from one spiral cycle to the next. Each choice of answers generates a different process model. Rather than develop the completed product in one step, multiple cycles are performed with each taking steps calculated to reduce the most significant remaining risks. Each anchor point milestone is a specific combination of artifacts and conditions that must be attained at some point. The sequence of three anchor point milestones are "LCO," "LCA," and "ICO". These milestones impel the project toward completion and offer a means to compare progress between one project and another. Many aspects of spiral development are omitted in the above definition. The remainder of this paper expands the definition by describing six essential aspects that every proper spiral process must exhibit. Each subsequent section describes a Spiral Essential, the critical success factor, reasons why it is necessary, and the variant process models it allows. Examples are given. Other process models that are precluded by the Spiral Essential are described. Because these may seem to be instances of the

spiral model, but lack necessary essentials and thus risk failure, they are called “hazardous spiral lookalikes.”

MOTIVATION TO SPIRAL MODEL



Software Engineers choose to model their problem spirally

- When creation of a Prototype is appropriate.
- When costs and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment unwise because of potential changes to economic priorities.
- Users are unsure of their needs.
- Requirements are complex.
- New Product line.
- Significant changes are expected (research and exploration).
- Evolving requirements can get accepted.
- Welcome use of prototyping.
- Feature requests have accurate definitions.

- Early product builds are available.
- Development can have subparts, and the development of the sensitive components happen earlier to minimize the risk involved.

The amount of risks in the spiral model is less when compared to other models because the errors or risks are identified and refined after the completion of each stage. Based on the risk patterns of a given project, the spiral model helps developers to increase the efficiency of the model as most risk is already handled.

Additional functionality or changes can be done at a later stage. Cost estimation becomes easy as the prototype building is done in small fragments. Continuous or repeated development helps in risk management. Development is fast and features are added in a systematic way. There is always a space for customer feedback.

APPLICATIONS

The Spiral Model is widely used in the software industry as it is in sync with the natural development process of any product, i.e. learning with maturity which involves minimum risk for the customer as well as the development firms.

Places where Spiral Models are used include –

- When there is a budget constraint and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
- Customers are not sure of their requirements which is usually the case.
- Requirements are complex and need evaluation to get clarity.
- New product line which should be released in phases to get enough customer feedback.
- Significant changes are expected in the product during the development cycle.

PROS AND CONS

The advantage of spiral lifecycle model is that it allows elements of the product to be added in, when they become available or known. This assures that there is no conflict with previous requirements and design.

This method is consistent with approaches that have multiple software builds and releases which allows making an orderly transition to a maintenance activity. Another positive aspect of this method is that the spiral model forces an early user involvement in the system development effort.

On the other side, it takes a very strict management to complete such products and there is a risk of running the spiral in an indefinite loop. So, the discipline of change and the extent of taking change requests is very important to develop and deploy the product successfully.

The advantages of the Spiral SDLC Model are as follows −

- Changing requirements can be accommodated.
- Allows extensive use of prototypes.
- Requirements can be captured more accurately.
- Users see the system early.
- Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.

The disadvantages of the Spiral SDLC Model are as follows −

- Management is more complex.
- End of the project may not be known early.
- Not suitable for small or low risk projects and could be expensive for small projects.
- Process is complex
- Spiral may go on indefinitely.

REQUIREMENTS THEORY

FUNCTIONAL REQUIREMENTS:

The behavior of a system is defined by its functional requirements. Essentially, they are **what** the system does or must not do, and can be thought of in terms of how the system responds to inputs. Functional requirements usually define if/then behaviors and include calculations, data input, and business processes.

It allows the system to function as it was intended. Put another way, if the functional requirements are not met, the system will not work. Functional requirements are product **features** and focus on user **requirements**.

NON-FUNCTIONAL REQUIREMENTS:

While functional requirements define what the system does or must not do, non-functional requirements specify **how** the system should do it. Non-functional requirements do not affect the basic functionality of the system. Even if the non-functional requirements are not met, the system will still perform its basic purpose.If a system will still perform without meeting the non-functional requirements, why are they important? The answer is **usability**. Non-functional requirements define system behavior, features, and general characteristics that affect the user experience. How well non-functional requirements are defined and executed determines how easy the system is to use, and is used to judge system performance. Non-functional requirements are product **properties** and focus on user **expectations**.

CRITICAL NON-FUNCTIONAL REQUIREMENTS TO OUR SYSTEM

- **Security.** What are the security requirements, both for the physical installation and from a cyber perspective?
    - Details entered by the user are stored in a database that is secured by three-levels of protection.
    - They are transferred to the server using the POST method which encodes the data before sending it thereby ensuring the fact that hacking of sensitive user details turns out to be futile.
- **Reliability / Availability.** What are the uptime requirements? Does it need to function 24/7/365?
    - Application should be designed in a way to provide round-the-clock services to the users.
    - This is because the application functions for users from all around the globe whose timelines are substantially different.D
- **Scalability**. As needs grow, can the system handle it?
    - System needs to be designed in such a way that enormous workload could be handled with ease.
    - This is especially vital during holidays,festivals,etc when a huge number of people could potentially use the application.
- **Performance.** How fast does it need to operate?
- **Supportability**. Is support provided equally in all platforms?
    - Should be cross-platform as users from different environments can utilize the functionalities of the system with ease.
- **Usability**. This focuses on the appearance of the user interface and how people interact with it. What colour are the screens? How big are the buttons?
    - UI should be constructed in an appealing manner so that the user finds it interesting and interactive.

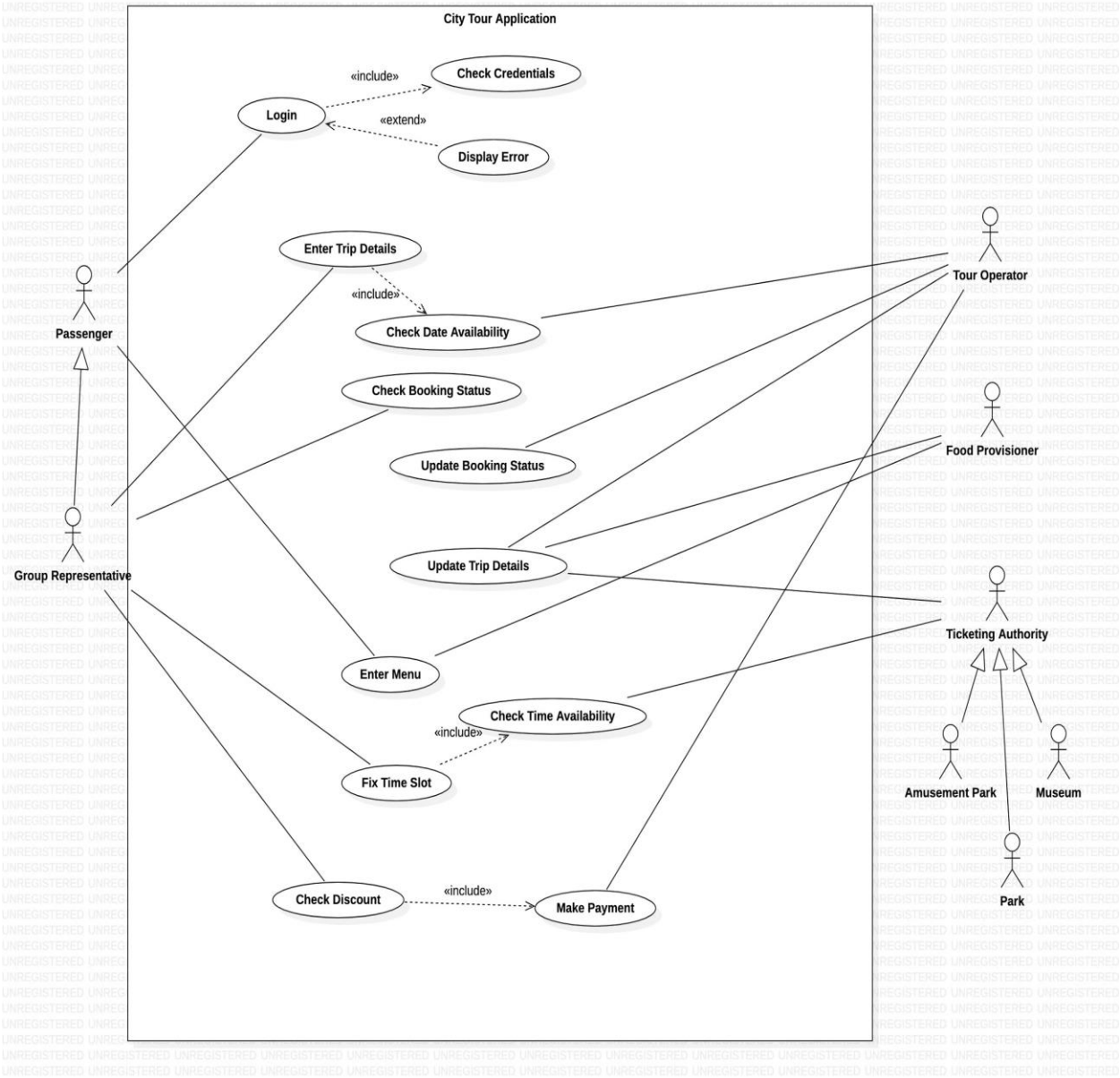
ITERATION 1 (INNER SPIRAL)

FUNCTIONAL REQUIREMENTS ELICITATION:

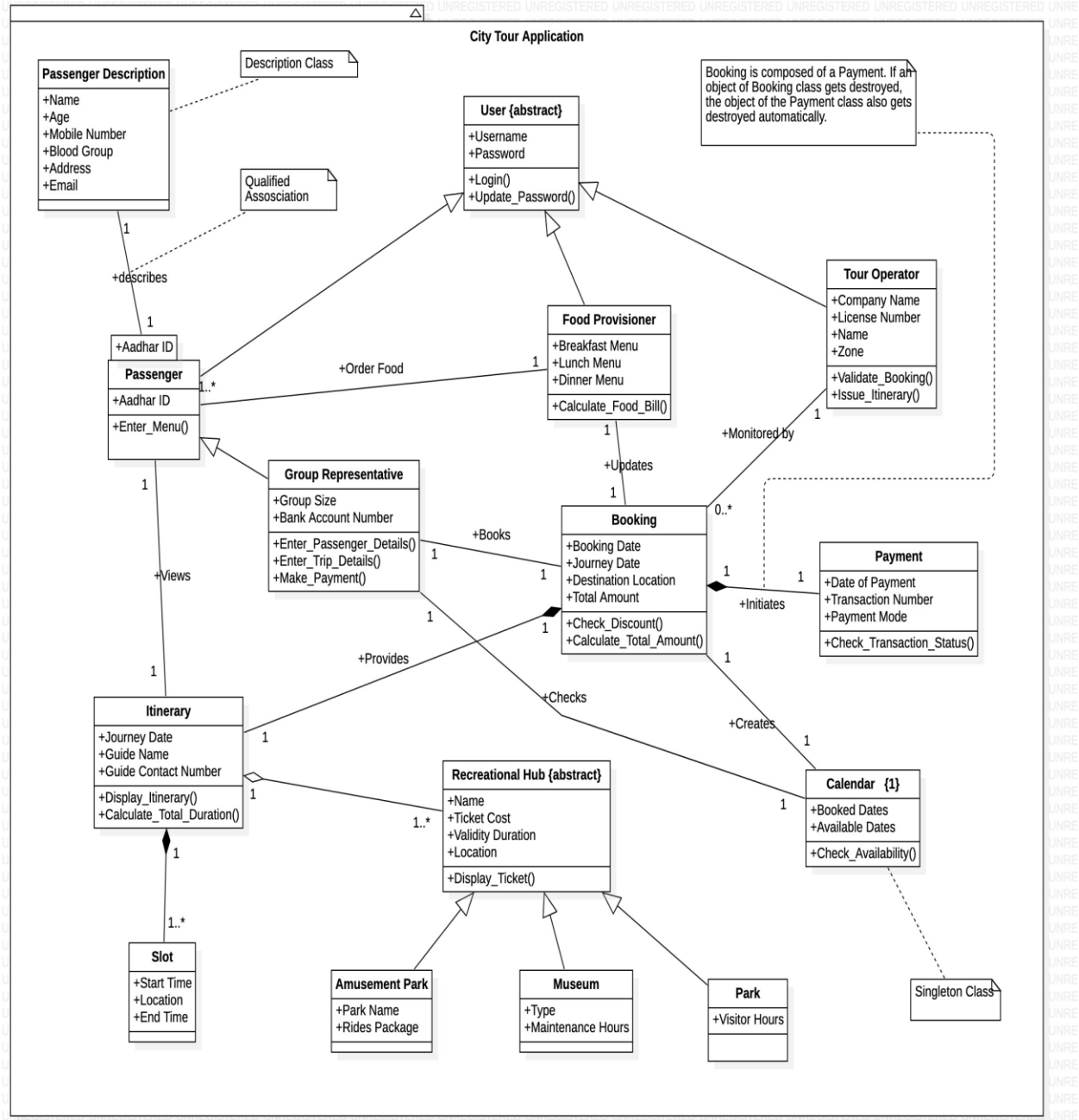
- User requires a functionality that enables him to:
    - Store his booking details.
    - Persistence of data, if the user changes phones or goes on to the desktop version of the website they will still have their search history, favorites, etc.
    - Receive personalized recommendations, newsfeeds, promotional offers and festival discounts.
    - Be more accountable for his actions with the aid of a well cultivated account on the line.
    - Retain his refined searches, settings and relationships built.
- Providing a plethora of tourist spots enables the user to plan his vacation with his near and dear ones at his own pace and comfort, avoiding hectic queues to make bookings as it used to be previously. Choice of the holiday destination can be made after surfing the catalogue provided.
    - Hence, the system requires a feature that should allow the user to enter his trip related details and his preferences so that the Tour Operator can arrange accordingly.
- The Tour Operator can handle only 50 people in order for the trip to be effective and enjoyable.
    - To take this into account, the system is in need of a functionality that checks this criteria once the user enters the trip details.
- User needs to be constantly updated about the current state of his booking so that he knows what's going around.
    - This is vital to improve customer experience.
- System requires a functionality such that when the Group Representative of a booking adds a passenger to the current lot, Booking ID gets updated in the database.

SYSTEM DESIGN:

USE CASE DIAGRAM



CLASS DIAGRAM:



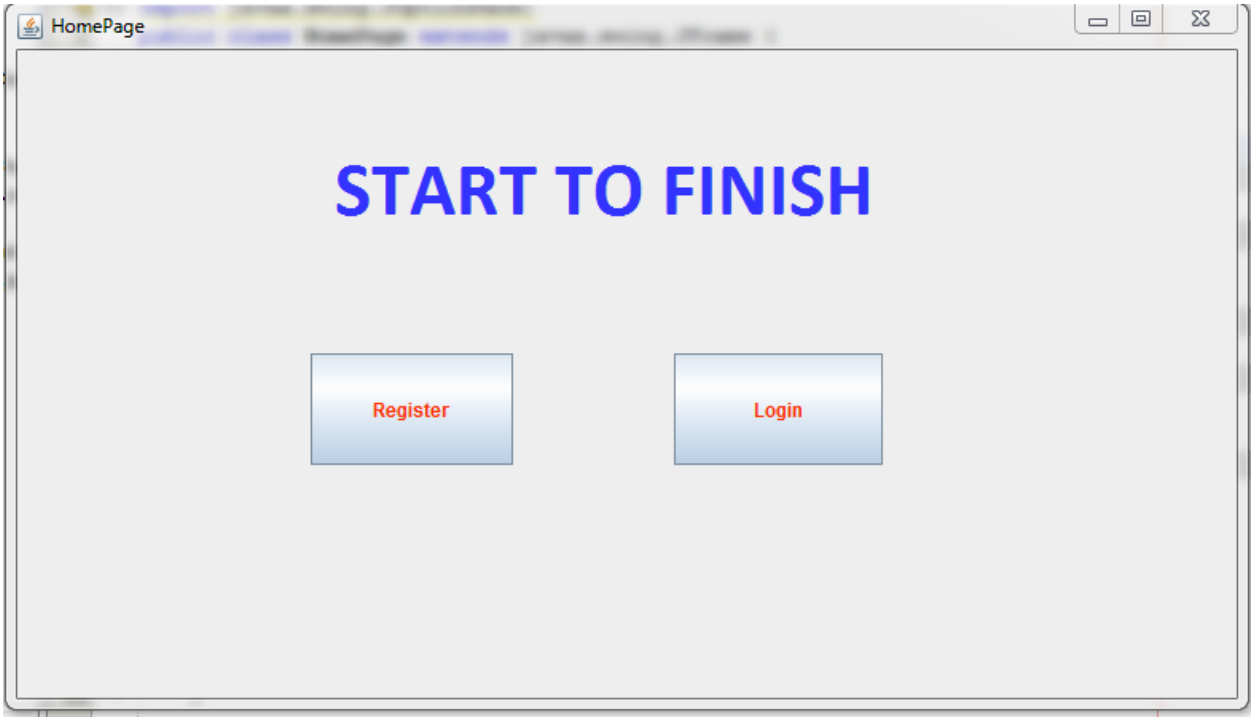
- Generalization:
    - Passenger, FoodProvisioner and TourOperator classes are subclasses of User Class which is an abstract class.
    - GroupRepresentative class is in turn a subclass of Passenger class.
    - AmusementPark, Museum and Park classes are subclasses of the abstract class - RecreationalHub
- Composition:
    - Booking class composes the Itinerary
    - Booking class composes the Payment
    - Itinerary class composes the Slot
- Aggregation:
    - Itinerary class aggregates the RecreationalHub class.

Element	**Type**	**Functionality**
Login	Button	On clicking Login, the User gets redirected to the Login Page
Register	Button	On clicking Register, the User gets redirected to the Register Page
Email ID	Text Field	Helps user to enter Email ID
Name	Text Field	Helps user to enter Name
Password	Password Field	Helps user to enter Password
Type of User	Combo Box	Helps users to select one of Passenger, Group Representative,Tour Operator from a dropdown list
Submit Register	Button	On successful submission, entered details are populated onto the database and user is redirected back to the home

		page
Submit Login	Button	On entering valid credentials, the user is taken to his corresponding dashboard. In case of invalid credentials, an error message pops up and he is prompted to enter his details again.
Back	Button	Redirects users back to the home page.
Make Booking	Button	Redirects to the Booking page.
Journey Date	Text Field	Helps the Group Representative to enter the date of journey.
Booking Date	Text Field	Helps the Group Representative to enter the date of booking.
Group Size	Text Field	Helps the Group Representative to enter the size of the group.
City	Text Field	Accepts the city to which they plan the tour.
Submit Booking	Button	Submits the entered details.
Logout	Button	Helps users log out of the system.

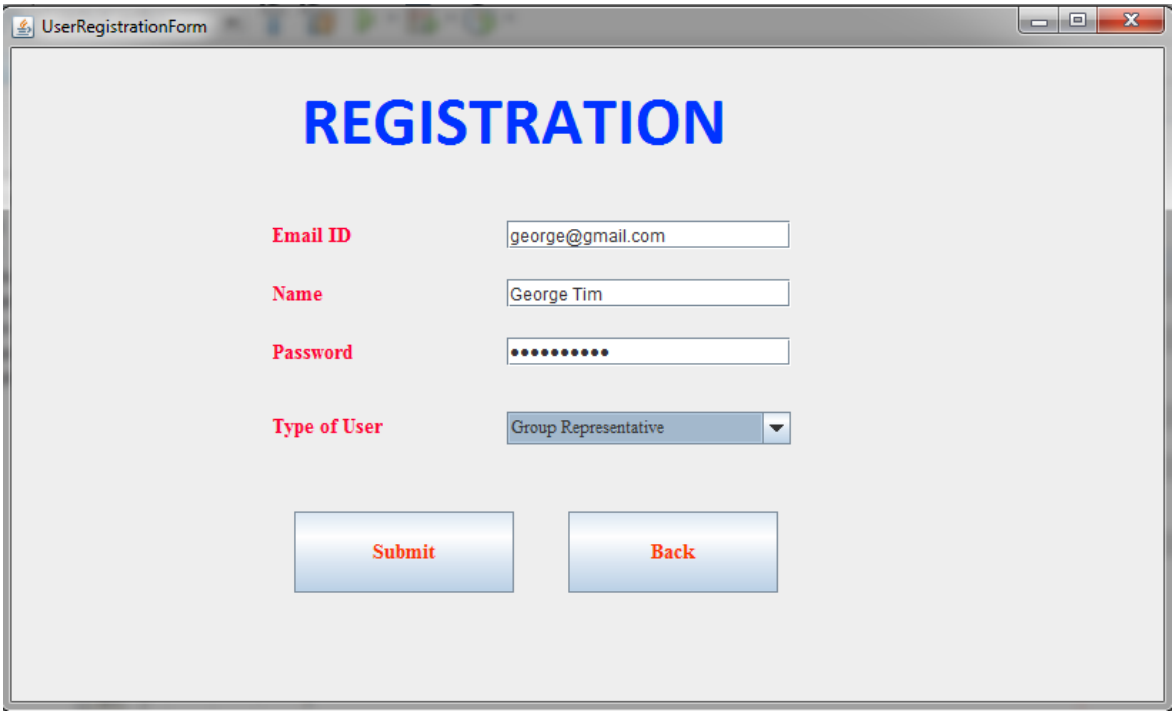
CONSTRUCT AND BUILD:

1) HOME PAGE



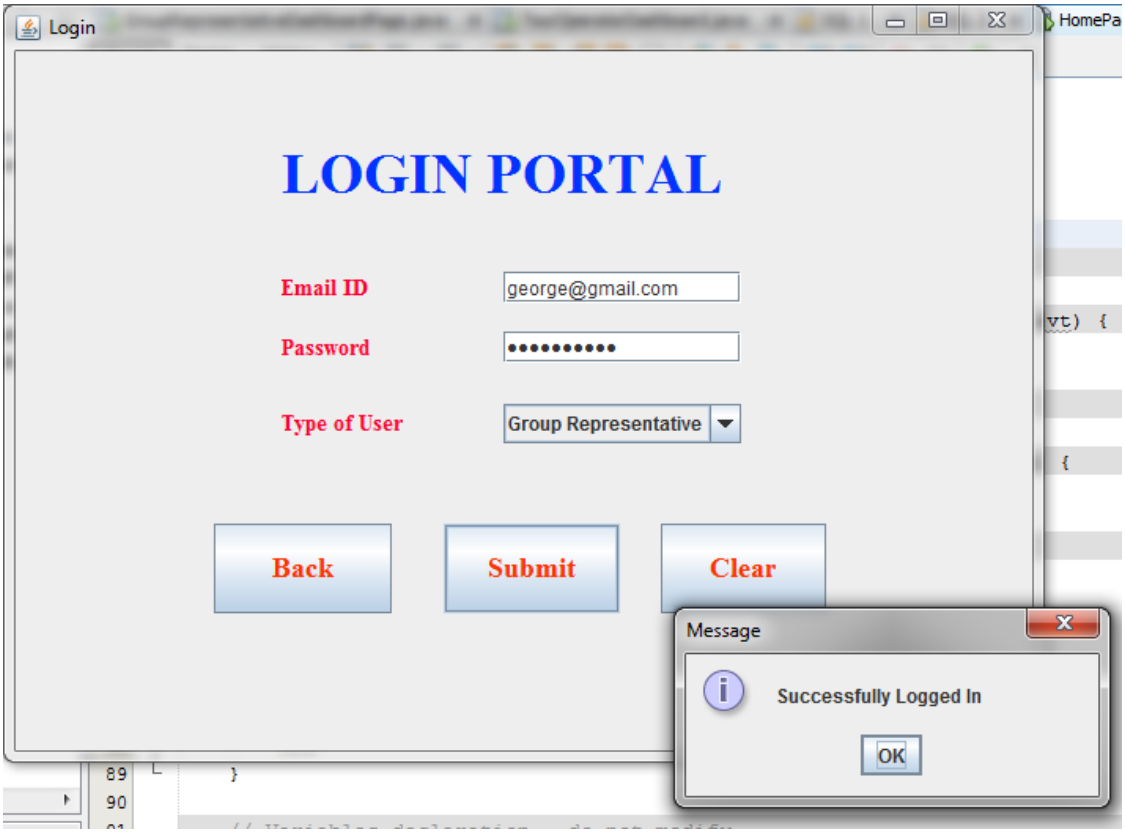
Initially the Home Page of the application is built which is what pops up when the application is opened. Having greeted the user into the application, this page has the provision to take the user to the Register page, where the user needs to create a registration inorder to avail the services provided. Login button takes the user to the Login page which acts as an intermediary to the dashboard if appropriate credentials are entered by the user as registered.

2) USER REGISTRATION PAGE



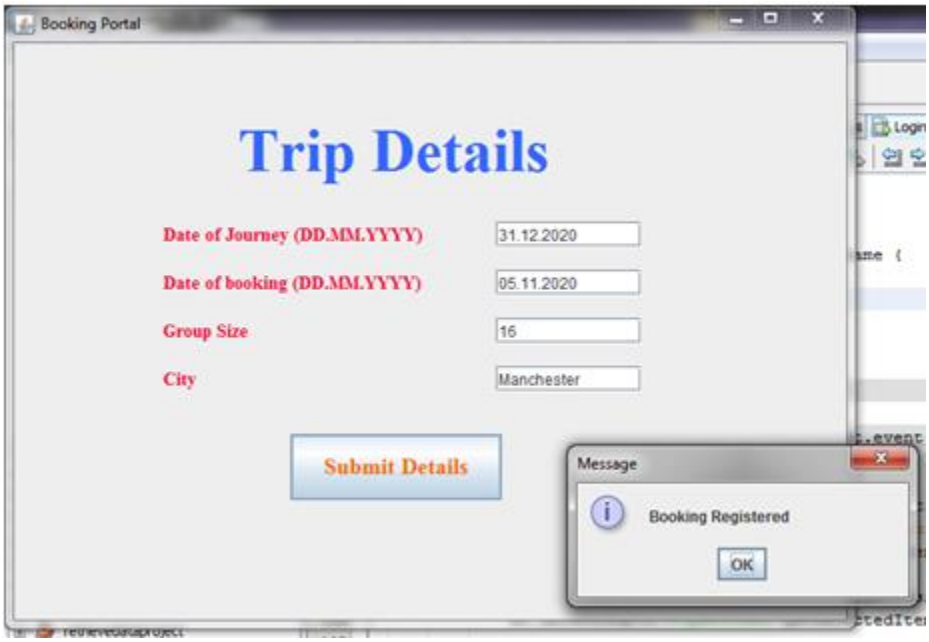
Inorder to avail the services of the system, it is mandatory that the user needs to be registered. The registration process is accomplished with the help of this User Registration Page. The details which are to be submitted by the user includes his/her email-ID, Name, Password and the Type of User. There are different types of users in this system like Passenger, GroupRepresentative each with their own set of functionalities.

3) LOGIN PAGE



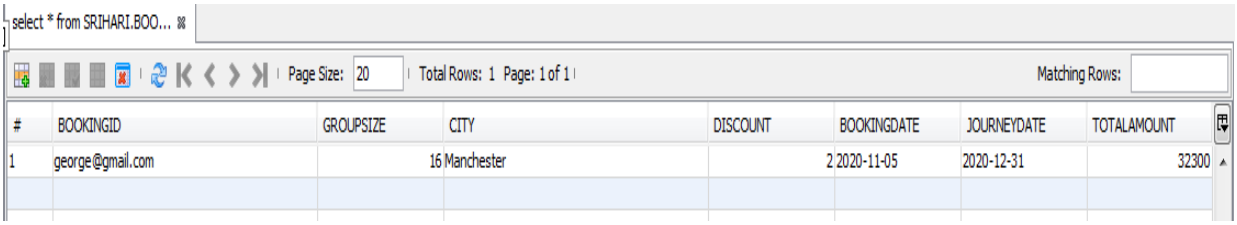
Having registered, the next step which is to be taken by the user to avail the services provided by the system is to Login. This is achieved by using the Login Page where the user needs to enter the details as requested. Only if the details provided by the user in this page match with those provided during registration, the user is taken to his dashboard.

4) BOOKING PORTAL



At first the booking needs to be done by entering the overall details like the Date of Journey, Date of Booking, Size of the Group travelling and their dream destination. This can be done only after logging in. The privilege of making a booking is provided only to the GroupRepresentative. The Submit details button saves this entry in the BookingDetails table and throws a pop-up to the user indicating “Booking Registered”.

BOOKING_DETAILS_TABLE



#	BOOKINGID	GROUPSIZE	CITY	DISCOUNT	BOOKINGDATE	JOURNEYDATE	TOTALAMOUNT
1	george@gmail.com	16	Manchester	2	2020-11-05	2020-12-31	32300

The Booking Details table has fields like BookingID, GroupSize, City, Discount, BookingDate, JourneyDate and TotalAmount as shown above. On clicking the submit details button in the Booking Portal the required fields are populated in the database.

EVALUATION AND RISK ANALYSIS

TECHNICAL FEASIBILITY

During the initial iterations, evaluating and analysing the risks involved with the technical aspects of the software is an integral part in proceeding further, as it is via the digital model that we are able to reach out to our customers. So we tried to ask ourselves the following questions:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipment have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of a secure infrastructure implementation system. The current system developed is technically feasible. Thus it provides easy access to the users. The database's purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified.Therefore, it provides the technical guarantee of accuracy, reliability and security. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing a fast feedback to the users irrespective of the number of users using the system.

ECONOMIC FEASIBILITY

In Economic Feasibility, the cost and benefit of the project is analyzed. Means under this feasibility study, a detailed analysis is carried out of what will be the cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analyzed whether the project will be beneficial in terms of finance for organization or not.

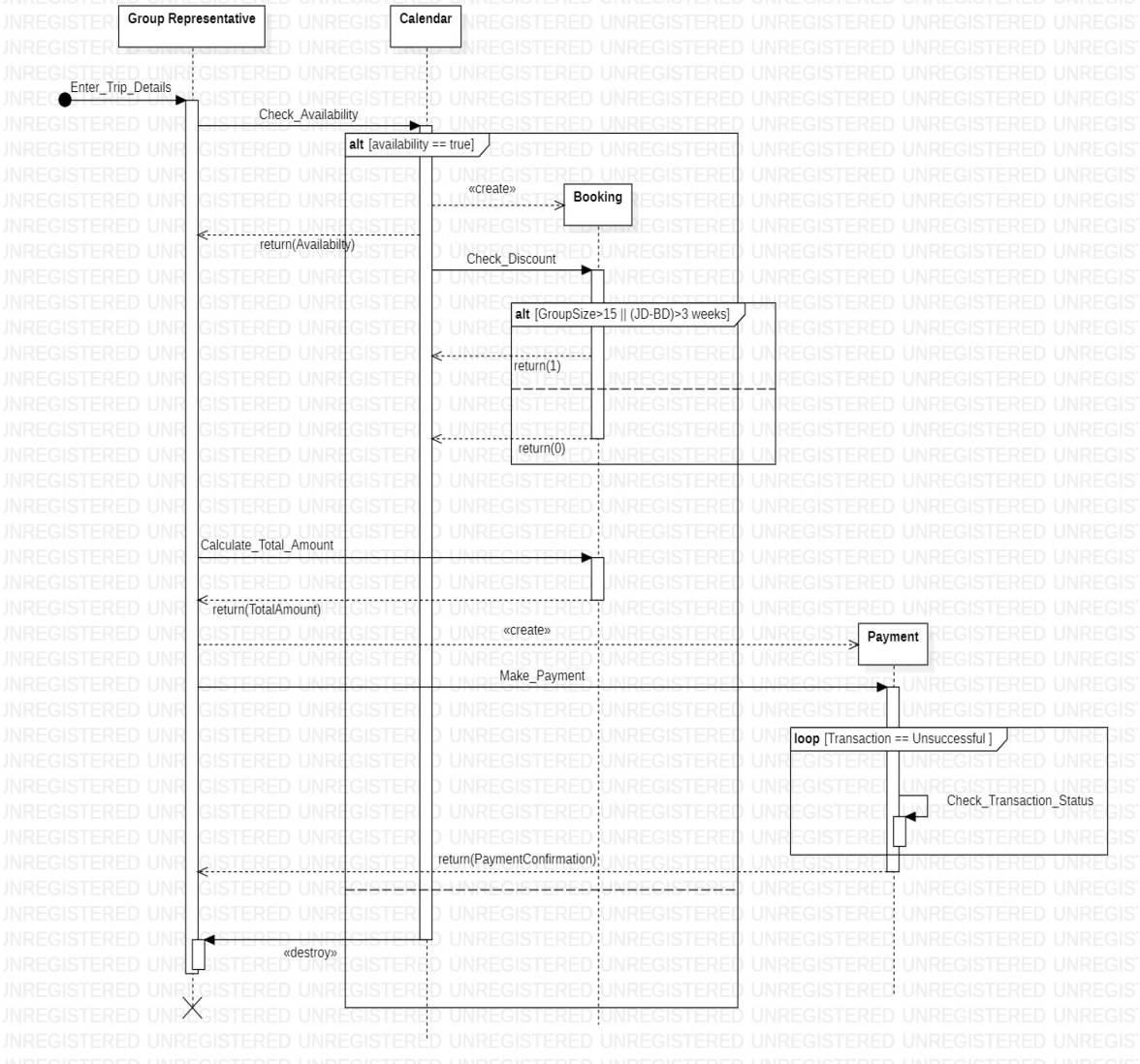
ITERATION 2 (MIDDLE SPIRAL)

FUNCTIONAL REQUIREMENTS ELICITATION:

- Users generally don't have time to do the necessary research about the city of visit or they simply don't know where to start. A great guide can completely transform a trip, making a destination come alive with historical anecdotes, specialized visits and insider access.
    - System requires a functionality that gives the users an option of choosing a Tour Guide.
- Each user has a different lifestyle and a specific foodstyle and we are entitled to provide them a personalized experience.
    - System requires a functionality that allows the user to choose from a variety of cuisines at a nominal cost so that the concept of “Food” is never a problem.
    - The user is charged depending on his choice of cuisine and the corresponding cost is updated.
- To attract customers to us, a wide range of discounts are to be provided to the users.
    - Discount Coupons are to be provided to frequent loyal customers so that they get attractive offers on recommending us to their acquaintances.
    - Special discounts are to be provided for advanced and bulk bookings.

- Making Payments has always been a tedious and time consuming process from time immemorial.
    - System requires a functionality that allows a wide range of payment options to the user.
    - Payment process needs to be client-centric. Offer a variety of tools and payment options to accommodate customer needs.
    - Ensure cross-functionality for your payment processes.
    - Make payment processing easy for clients by using a simple interface and features like guest checkout.

SYSTEM DESIGN - SEQUENCE DIAGRAMS



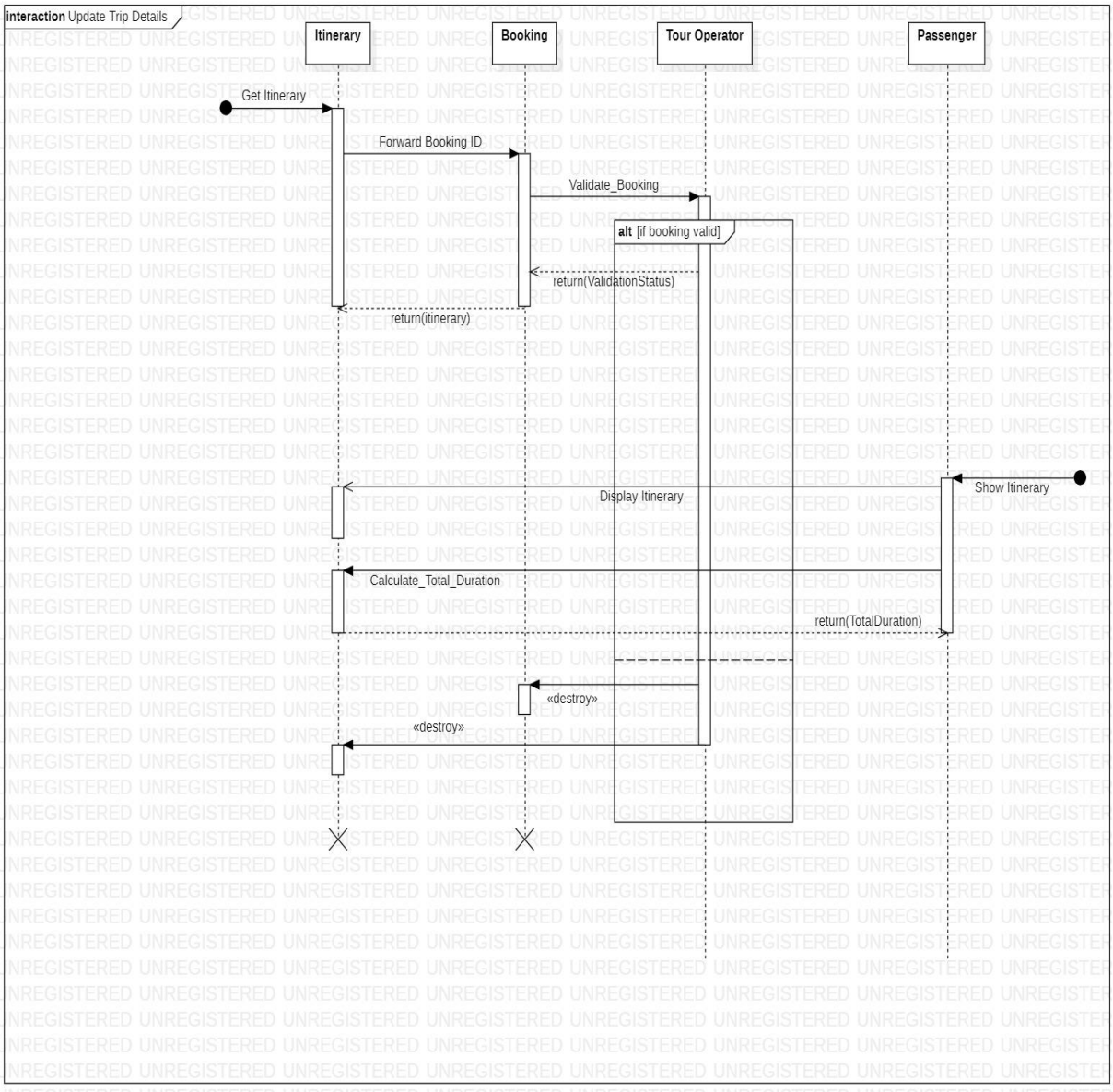
1. While entering the TripDetails the GroupRepresentative checks for the available dates in the calendar.

2. If the requested date is available for booking, he/she creates a booking and checks for discounts if any.

3. Discounts are provided if the GroupSize is more than 15 or if the booking is made 14days in advance.

4. The total amount is then calculated and the payment is made.

5. The transaction status is checked until it is successful.



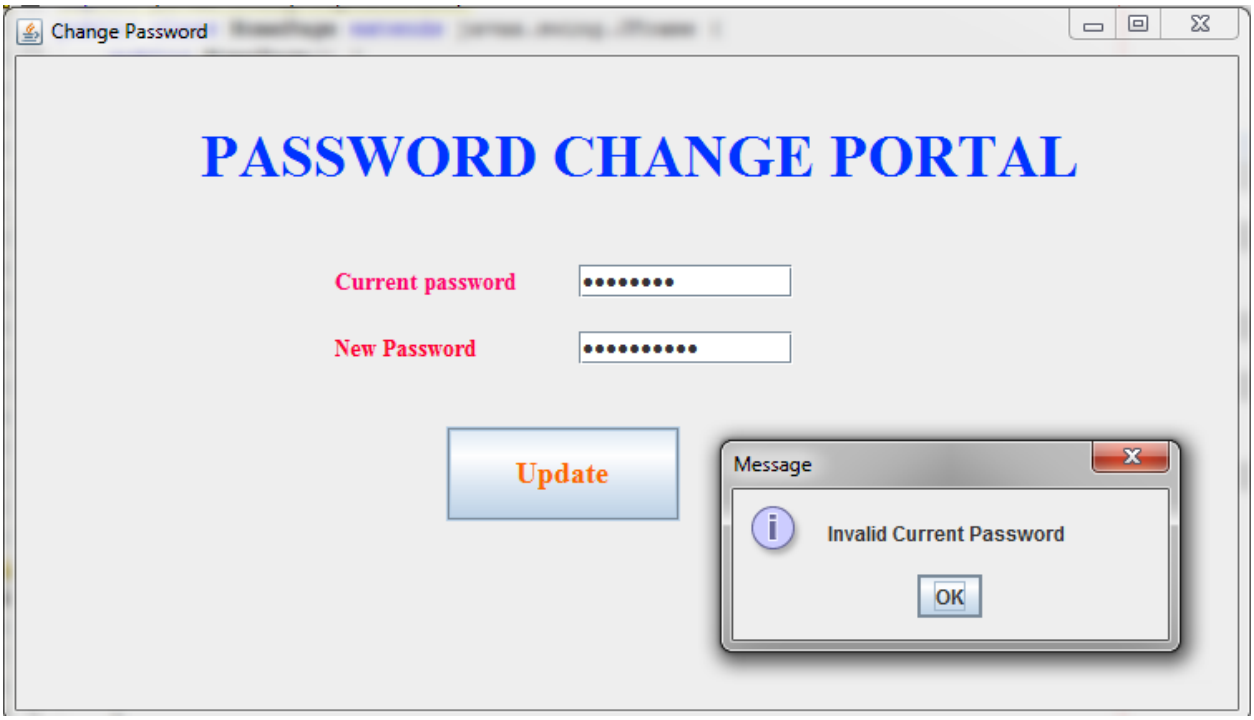
1. To generate the itinerary for the passengers, the booking ID is forwarded to the TourOperator.
2. TourOperator validates the booking.
3. An itinerary is generated if the booking is valid.
4. If invalid no itinerary gets generated for the passenger.
5. When the passenger requests for the itinerary, it is displayed.
6. The total duration of the trip can be found from the itinerary.

Element	**Type**	**Functionality**
Passenger Details	Button	Enables him to add passengers to the current booking.
Add Passenger	Button	Adds a passenger to the booking(Booking ID field of Passenger gets filled appropriately).
Submit Group	Button	Submits the passenger details and redirects to the Group Representative's dashboard.
Update Menu	Button	Food preferences given by the user and the corresponding costs are updated in the database.
Choose Menu	Combo Box	Once booking is done, the user is given an option to provide his food preferences(Veg, Non-Veg, Combo).
Apply Discount	Button	Calculates discount based on advanced booking/bulk booking (if applicable) and updates the cost in

		the database.
Make Payment	Button	Helps the user to make payment and change the booking status in the database from 0 to 1.
Enter Current Password	Password Field	Helps the user to enter his valid current password
Enter New Password	Password Field	Helps the user to enter his new password
Update Password	Button	• On entering a valid,synchronising current password,new password pair, new password is updated in the database and the user is redirected to the home page. • On entering invalid details, he is prompted to enter valid passwords and is given a chance to enter them again.

CONSTRUCT AND BUILD:

1) CHANGE PASSWORD PAGE



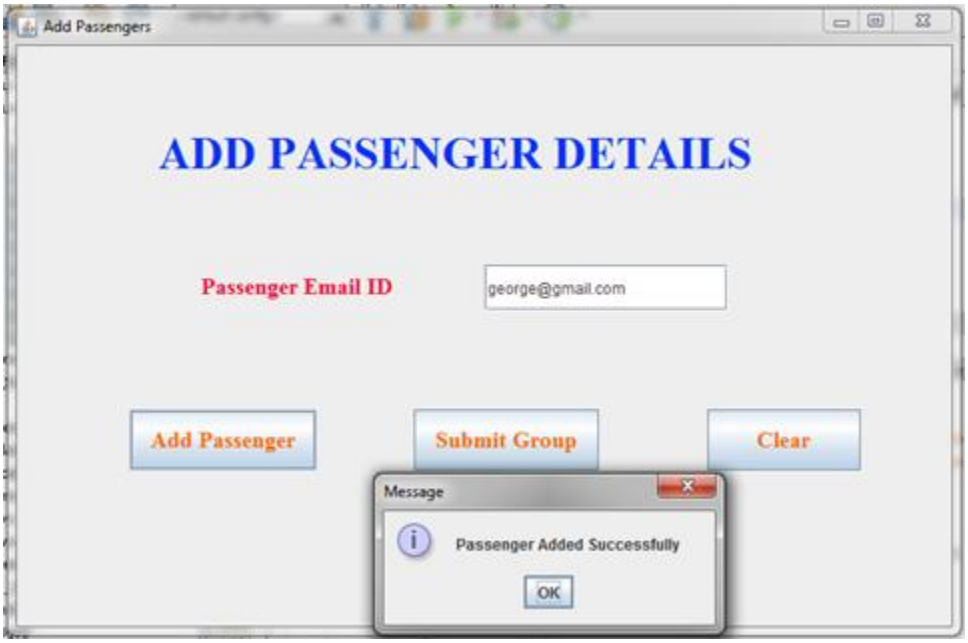
Sometimes an User might spend sleepless nights feeling insecure if his/her password has been hacked or is vulnerable to. To avoid all that, the ChangePassword page enables him/her to set a new password. To achieve this, initially the current password needs to be entered followed by the new password. The newly set password would be updated only if the valid current password is entered, thereby ensuring authenticity. The Update button helps to get things done. In case the wrong current password is entered, the user is prompted with the message “Invalid Current Password” and is provided another opportunity to reset the password.

2) GROUP REPRESENTATIVE DASHBOARD

Majority of the powers with regards to the booking lies in the hands of the GroupRepresentative. Upon logging in he/she is taken to the dashboard where a number of services can be availed. One of these include the MakeBooking feature by which the GroupRepresentative can enter the trip details to affirm a booking. This is done through the MakeBooking button.

The passengers who would be onboard in the trip are added to the booking by the GroupRepresentative using the Add Passenger Details button. Eye-catching discounts are provided to bookings on a number of criteria. Discount upto 10% of the total cost is provided for booking with group size more than 15, which 20% discounts are provided if the booking is made a fortnight before the travel date. The total cost calculated upon applying discounts if any is shown to the GroupRepresentative if the Apply Discount button is clicked. Simultaneously the cost is updated in the database. Pressing the Make Payment button makes the payment for the booking and changes the booking status from 0 to 1 indicating that it is ready for validation.

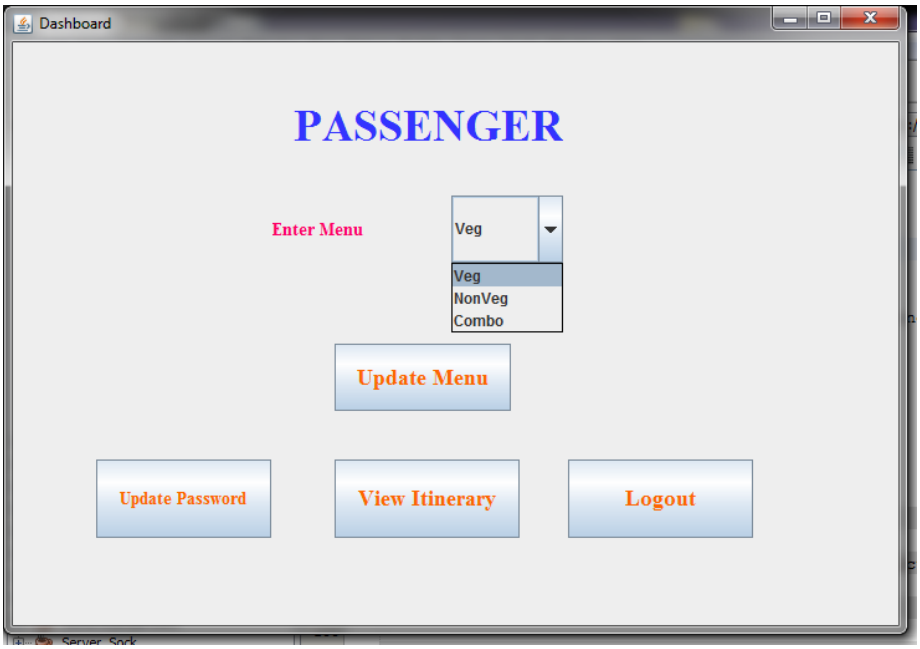
3) ADD PASSENGERS PAGE



With the GroupRepresentative rests the huge responsibility of adding co-passengers to the trip. Inorder to do this, the Add Passengers Page enables him/her to enter the email id of the passenger in a textfield. Upon clicking the Add Passenger button, the corresponding passenger is added to the GroupRepresentative's trip. It is greeted with a pop-up "Passenger Added Successfully". If he/she wishes to erase the email id entered, the Clear button can be made use of. Submit Group indicates the end of the passenger list for the booking and takes the GroupRepresentative back to the dashboard.

4) FOOD PREFERENCES

Once passengers are added to a booking they are now allowed to enter their food choices which are provided in their dashboard. They are allowed to choose any one of Veg, Non-Veg and Combo. The cost gets updated in the database as Rs.10000,Rs.12000 and Rs.11000 respectively. These are in turn added to the overall cost of the booking.



USER DETAILS TABLE

#	EMAIL	NAME	PASSWORD	BOOKINGID	FOOD
1	george@gmail.com	George Tim	george tim	george@gmail.com	NonVeg
2	gokul@yahoo.com	gokul	abcde	george@gmail.com	Veg
3	srihari	hari	hari	george@gmail.com	NonVeg

ITINERARY	AMOUNT	BOOKINGSTATUS	TYPEOFUSER
Manchester-31.12.2020	12000	2	Group Representative
Manchester-31.12.2020	10000	2	Group Representative
Manchester-31.12.2020	12000	2	Group Representative

The user-details table shows the details of the registered users along with their food preferences and its cost, booking id, booking status and the itinerary once it is generated.

EVALUATION AND RISK ANALYSIS:

OPERATIONAL FEASIBILITY

Proposed projects are beneficial only if they can be turned into information systems that will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues we raised were to test the operational feasibility of the project and it includes the following: –

- Is there sufficient support for the users from the management ?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

Some of these risks from our system's perspective could be

- There are possibilities by which the user might wish to cancel his booking and request for refund.
- In such cases, the management should be responsible enough to consider the customer's emergency situations and refund back a major proportion of the booking amount.
- In case of a failed transaction, adequate measures should be taken in advance so that the amount deducted from the customer's bank account is completely refunded back within 24 hours from booking.

ITERATION 3 (OUTER SPIRAL)

FUNCTIONAL REQUIREMENTS ELICITATION:

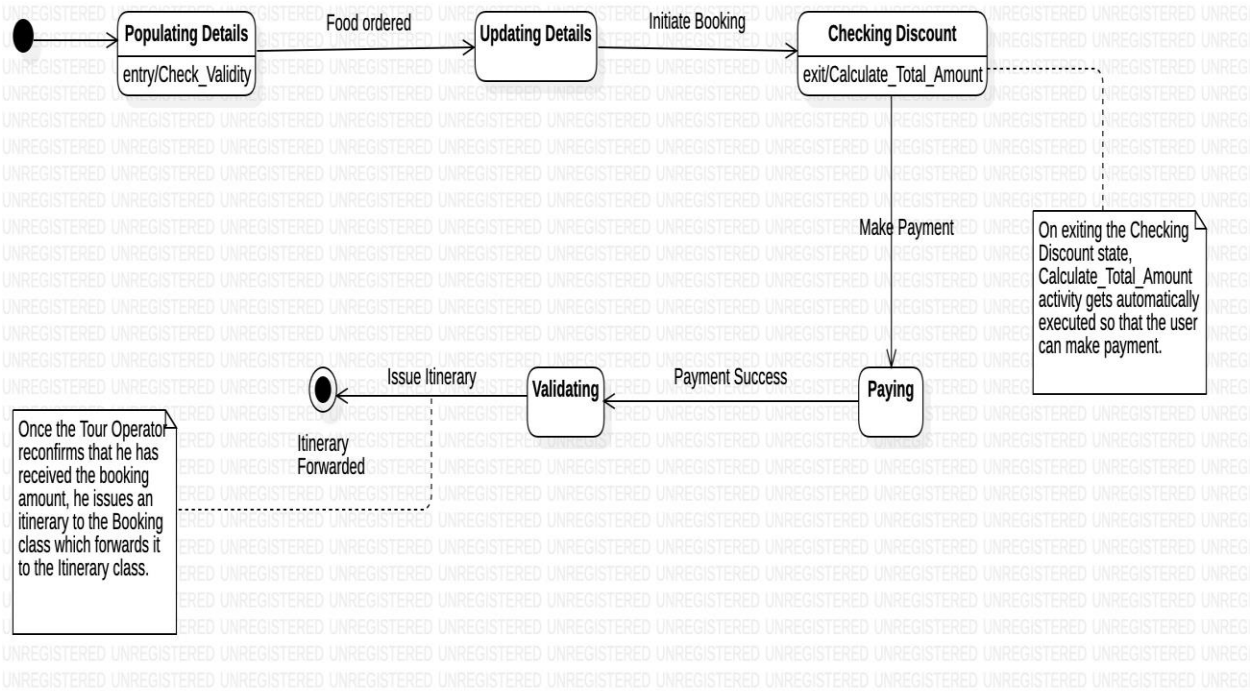
- A quintessential feature for any system of our sorts is fraud detection and validation.
    - System requires a functionality that allows our ambassador to validate a booking on the payments front as well as the user's authentication front.
- Itinerary is important because it can help maximize the customer's time during the trip, also helping him decide what activities he thinks he can afford to do during the trip.Itinerary is also good for travelers who cannot truly manage their time. They don't have to plan every second of their trip by doing activities.
    - For every valid booking, a carefully designed itinerary is generated upon consulting the representatives of each place of visit(amusement parks, museums,etc).
    - This enables us to synchronise the customer's time of visit with that of the working hours of the place of visit so that frequent hassles can be prevented.
- Each customer has a different taste when it comes to travel and it may or may not go hand-in-hand with that of his fellow passengers.

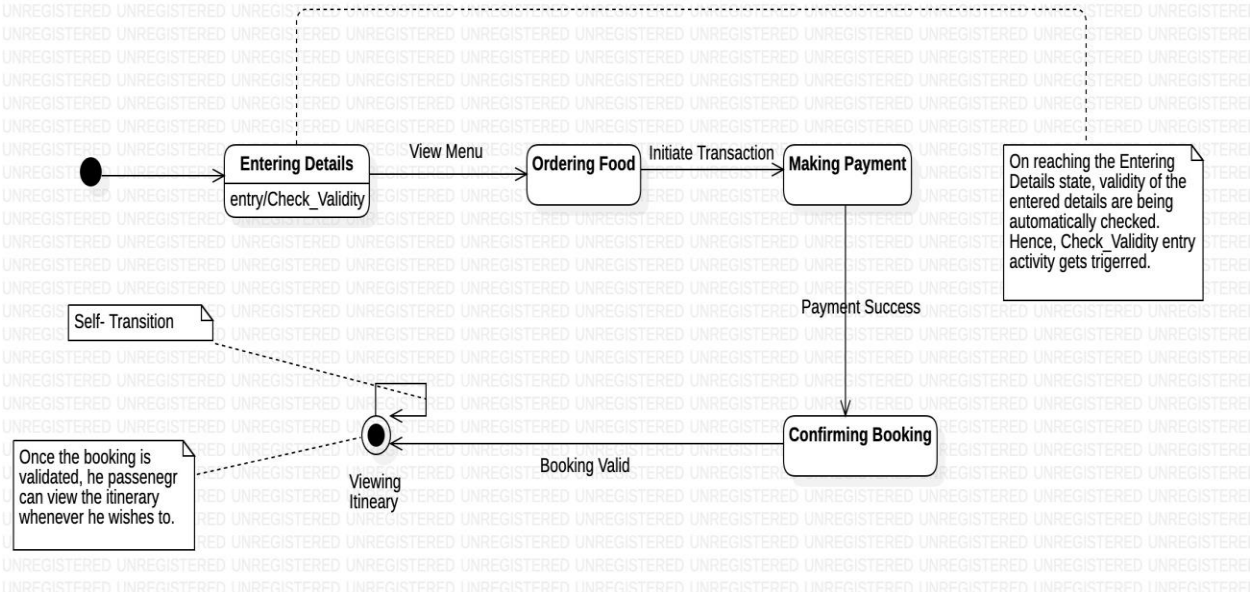
- When it comes to amusement parks/museums, the system is in need of a functionality that should allow the customer to enter his interests(preferred sections of the museum/park) so that he whole-heartedly enjoys his trip.
- Once he provides his choice, the system automatically generates a bill so that the user can make the required payment.
- Itinerary should also be updated accordingly.

- Transportation has been an integral part of the Tourism industry.It links tourists with various tourist attractions. Each user prefers a different type of vehicle for his trip(Car - SUV, Sedan,Hatchback / Bus / Van/ Driverless Car(Zoom))
  - System requires a functionality to accept the user's transportation preferences and update the corresponding cost into his final bill.
  - Once successful payment has been made, driver details are to be sent to the user so that they can be in contact beforehand.
- Any organization needs to make its customers feel important and involved.By asking them to provide feedback, we're communicating to them that their opinion is valued.Customers feel important because we're treating them as such and they feel involved in shaping the product.
  - System requires a functionality that enables users to provide their valuable feedback with respect to the overall experience.
  - This is extremely important to constantly update our service standards and stay alive in the market.

SYSTEM DESIGN:

STATE DIAGRAMS

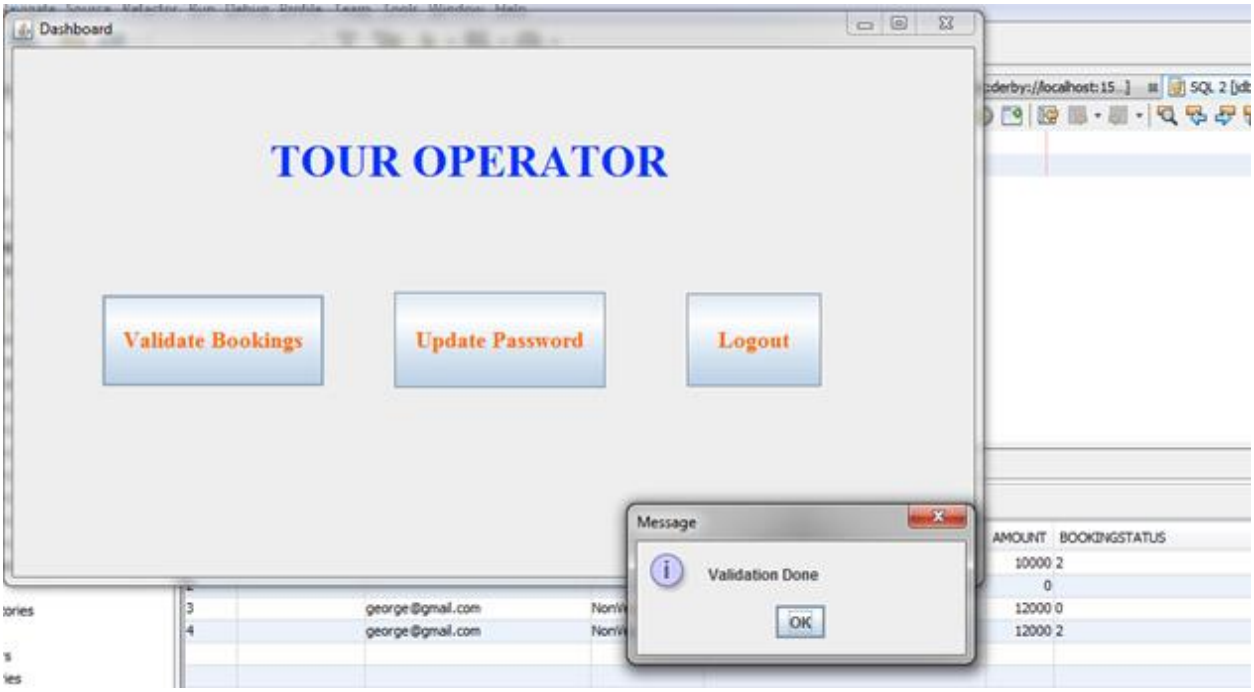




Element	Type	Functionality
Validate Bookings	Button	Helps the Tour Operator to validate a booking by changing its status from 1 to 2 thereby enabling the View Itinerary functionality to the Passenger.
View Itinerary	Button	Itinerary gets displayed to the Passenger once the Tour Operator validates the booking.
Customer Feedback	Text Field	Helps the user to rate the overall service experience and provide future improvement suggestions.
Submit Feedback	Button	Updates the submitted feedback onto the database.
Mode of Transport	Combo Box	Helps the user to choose his preferred mode of transport from a dropdown list.
Submit Transport	Button	• Updates the user's transport preferences onto the database. • Corresponding cost gets added to his overall bill.

CONSTRUCT AND BUILD:

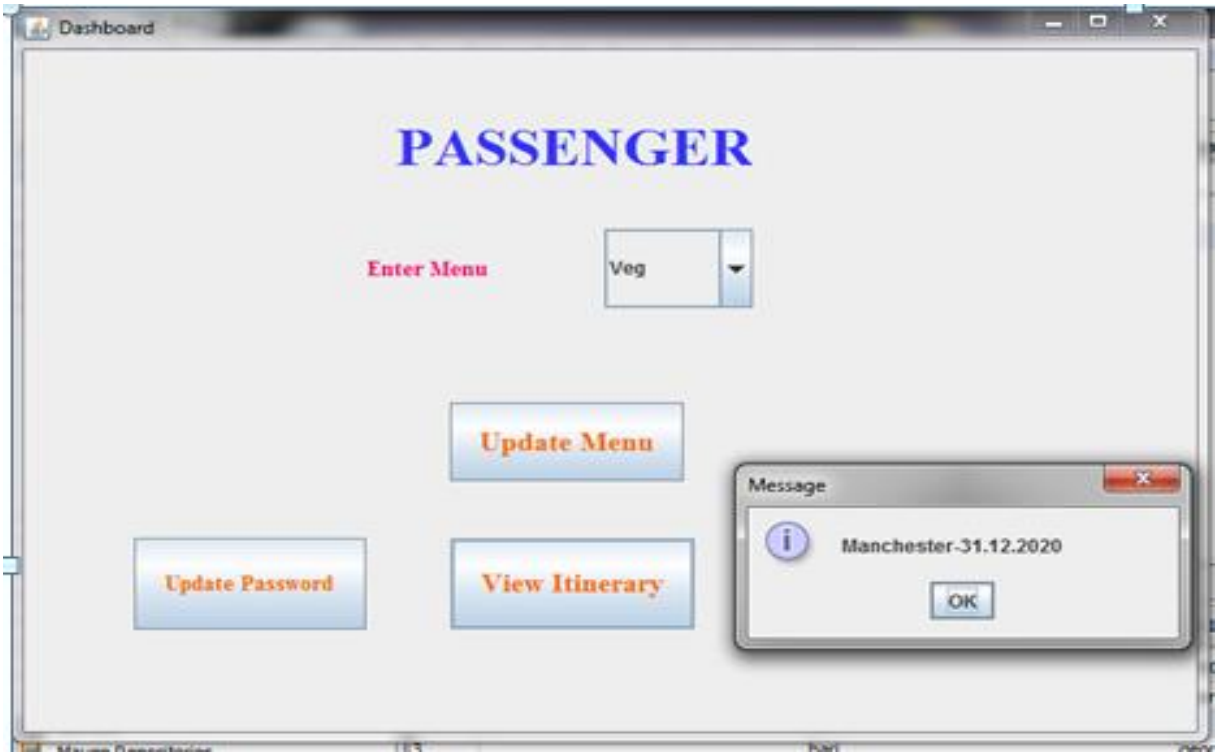
1) TOUR OPERATOR DASHBOARD



A representative from the organization appointed to monitor the bookings and take care of fraud detection. This official is another type of user of the system and is categorized as the Tour Operator. His credential formalities are alike as those of passengers. The main job of the TourOperator is to validate the bookings for which payment has been done. This is achieved by using the ValidateBookings button in his dashboard. For indicating the success of this process, he/she is provided an acknowledgement from the system - “Validation Done”.

2) ITINERARY GENERATOR

Every booking that has been validated by the Tour Operator, has an itinerary generated, which provides a clear sketch of the trip for the passengers. This allows them to plan their day accordingly. Until the booking is validated the View Itinerary button indicates the passenger that it hasn't been generated yet. But once validated the itinerary is provided as indicated.



EVALUATION AND RISK ANALYSIS:

LEGAL FEASIBILITY

The Legal Feasibility study project is analyzed in legality point of view. This includes analyzing barriers of legal implementation of project, data protection acts or social media laws, project certificate, license, copyright etc. Overall it can be said that a Legal Feasibility Study is a study to know if proposed projects conform legal and ethical requirements.

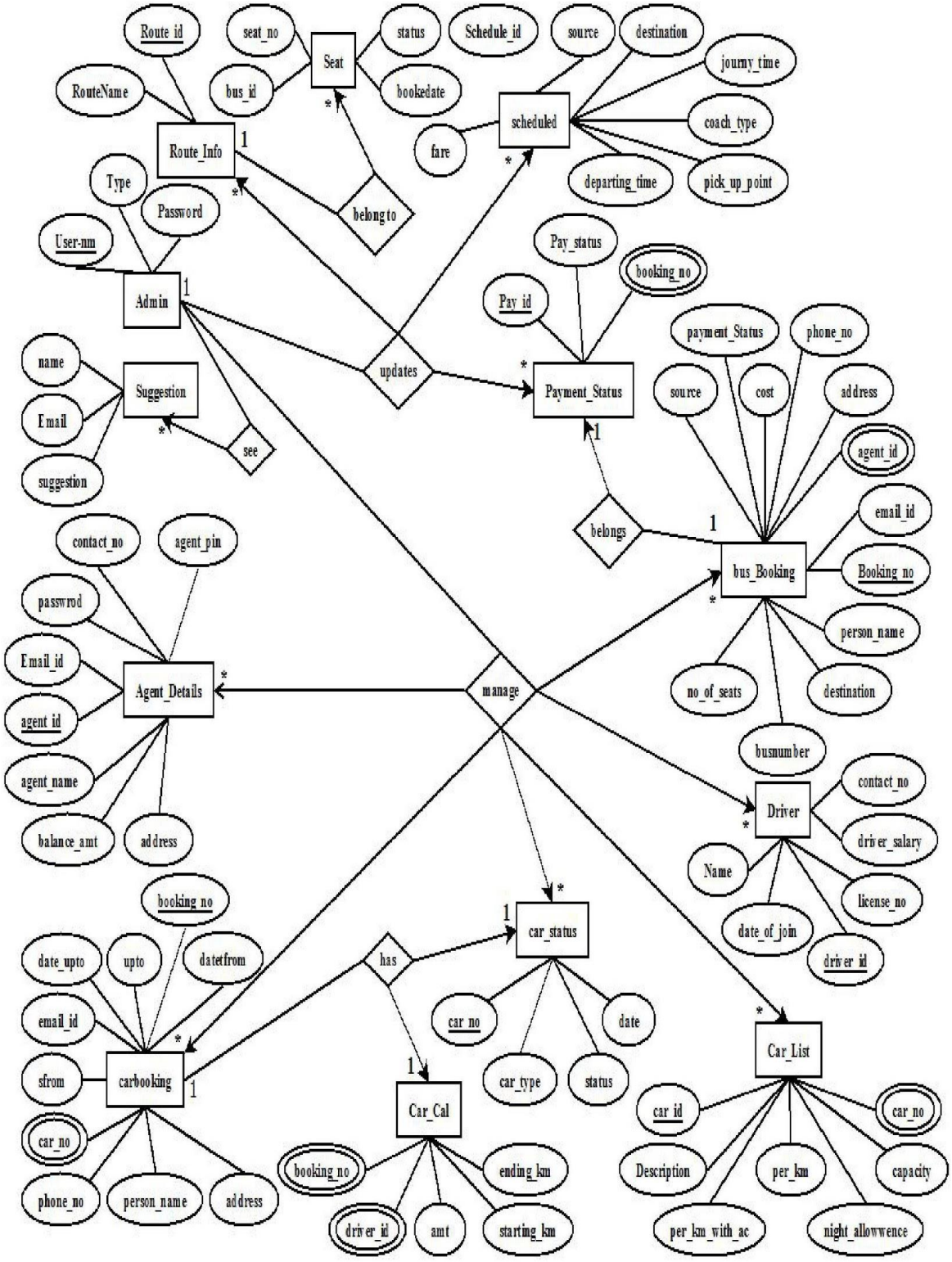
SCHEDULE FEASIBILITY

In Schedule Feasibility Study, mainly timelines/deadlines are analyzed for the proposed project which includes how many times teams will take to complete the final project which has a great impact on the organization as the purpose of the project may fail if it can't be completed on time.

CURRENT SYSTEM AND LIMITATIONS:

- The size of the database increases day-by-day, increasing the load on the database backup and data maintenance activity.
- Training for simple computer operations is necessary for users working on the system.

E-R Diagram


Route id
seat_no
Seat
status
Schedule_id
source
destination
journy_time
RouteName
bus_id
bookedate
scheduled
coach_type
Route_Info
fare
departing_time
pick_up_point
Type
Password
belong to
User-nm
Admin
Pay_status
booking_no
Pay_id
payment_Status
phone_no
name
updates
Payment_Status
source
cost
address
Suggestion
Email
see
agent_id
suggestion
email_id
belongs
contact_no
agent_pin
bus_Booking
Booking_no
passwrod
person_name
Email_id
Agent_Details
manage
no_of_seats
destination
agent id
busnumber
agent_name
Driver
contact_no
balance_amt
address
driver_salary
Name
booking no
license_no
car_status
date_of_join
driver id
date_upto
upto
datetfrom
has
car_no
date
email_id
Car_List
sfrom
carbooking
car_type
status
Car_Cal
car_id
car_no
car_no
booking_no
ending_km
Description
per_km
capacity
phone_no
person_name
address
driver_id
amt
starting_km
per_km_with_ac
night_allowwence

FINAL DATABASE DEFINITION:

Number of Tables: 2

Table 1: User Details

Field	Datatype
Email	VARCHAR(60)
Name	VARCHAR(40)
Password	VARCHAR(40)
Amount	NUMERIC
Type of User	VARCHAR(20)
Booking ID	VARCHAR(60)
Booking Status	INTEGER
Food	VARCHAR(10)
Itinerary	VARCHAR(60)

Primary Key: Email

Foreign Key: Booking ID

Table 2: Booking Details

Field	Datatype
Booking ID	VARCHAR(60)
Date of Booking	DATE
Date of Journey	DATE
City	VARCHAR(60)
Group Size	INTEGER
Discount	INTEGER
Total Amount	NUMERIC
Rating	NUMERIC
Mode of Transport	VARCHAR(30)

Primary Key: Booking ID

SYSTEM SECURITY

The protection of computer based resources that includes hardware, software, data, procedures and people against unauthorized use is known as System Security.

System Security can be divided into four related issues:

- Security
- Integrity
- Privacy
- Confidentiality

SYSTEM SECURITY refers to the technical innovations and procedures applied to the hardware and operation systems to protect against deliberate or accidental damage from a defined threat.

DATA SECURITY is the protection of data from loss, disclosure, modification and destruction.

SYSTEM INTEGRITY refers to the power functioning of hardware and programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping.

PRIVACY defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

CONFIDENTIALITY is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection.

CONCLUSION:

Our application happens to be an important tool that tourists can use in a handy manner to easily and efficiently schedule their trips. It brings them in contact with some of the best recommendations that not only suit them personally but also happen to be the most cost effective one. The main purpose of the above proposed project is to save the time of the tourists and provide them with proper guidance and directions regarding the tour they are about to undertake. They are given a multitude of choices to choose from in every possible front (from food all the way till transportation) so that their interests could be matched efficiently. It comes with an interactive interface where the user can navigate back and forth between windows to cater to the functionalities they are currently in need of. To sum up, the long tedious process of physically waiting in ticket booking offices and on-spot facilitation centres at amusement parks and museums are eliminated once and for all with the introduction of the proposed system.

FUTURE ENHANCEMENTS:

In future, an offline version of the proposed system could be formulated so that it can be helpful for tourists in case they face adverse connectivity issues. Making it offline would increase its storage but it will be compromised as the tourist will get more benefits.This system being web-based and an undertaking of the Cyber Security Division, it needs to be constantly updated and continuously tested to check out for any security gaps.A console for the data centre may be made available to allow the personnel to monitor on the sites which were cleared for hosting during a particular period. In effect, the path for a long road has been laid by us and following it would greet developers with several interesting corollary systems related to our domain.

REFERENCES:

[1] Oriogun, P.K. A Survey of Boehm's Work on the Spiral Models and COCOMO II—Towards Software Development Process Quality Improvement. *Software Quality Journal* 8, 53–62 (1999).

[2] L. Wildman, C. Fidge and A. Rae, "A Spiral Model of Software Development and Enhancement with Fault Evaluation for Security-Critical Communications Devices" in Computer, vol. 21, no. 05, pp. 61-68, 2006.

doi: 10.1109/MC.2006.161

[3] D. Grier, "The Outward Spiral" in *Computer*, vol. 53, no. 04, pp. 73-76, 2020.

doi: 10.1109/MC.2020.2971402

url: https://doi.ieeecomputersociety.org/10.1109/MC.2020.2971402

[4] N. A. M. Maiden and G. Rugg, "ACRE: selecting methods for requirements acquisition," in Software Engineering Journal, vol. 11, no. 3, pp. 183-192, May 1996, doi: 10.1049/sej.1996.0024.

[5] Yanyu Tao and Daopin Cheng, "Research on personalized tour service system," 2012 International Conference on Information Management, Innovation Management and Industrial Engineering, Sanya, 2012, pp. 74-77, doi: 10.1109/ICIII.2012.6339735.

[6] R. Nurbadi, A. Amyus, B. Pratama, A. F. Kanz and A. N. Fajar, "Smart Travel System Based on Service Oriented Architecture," 2018 Indonesian Association for Pattern Recognition International Conference (INAPR), Jakarta, Indonesia, 2018, pp. 256-261, doi: 10.1109/INAPR.2018.8627026.

[7]J. Cheng, H. Hsiang and W. Wu, "The design of intelligent mobile tourism service system," 2010 International Computer Symposium (ICS2010), Tainan, 2010, pp. 813-817, doi: 10.1109/COMPSYM.2010.5685401.