**By Semester 5th of 3rd Year**

**M.Sc. (CA & IT)**

**Submitted By :- Sheth Priyansh (3153)**

**Shah Rutvi (3231)**

**Patel Swara (3128)**

**Group Id : - 56**

**Company Name : - Voculas Software Pvt. Ltd.**

**K.S.School of Business Management,**

**M.Sc.IT in Computer Application and Information Technology,**

**Gujarat University.**



Index

1. **Introduction**
   1. Organisation Profile
   2. Project Details
      1. Current system
      2. Proposed system
   3. Scope
   4. Objectives
2. **Requirement gathering**
   1. Stakeholder
   2. Requirement gathering technique used
   3. Consolidated list of requirement
   4. Project Definition
3. **Project Management & Planning**
   1. Feasibility study
      1. Technical
      2. Economic
      3. Operational
   2. H/W S/W Requirement
   3. Project Planning
      1. Work Breakdown structure
      2. Gantt Chart
   4. Project Model
4. **Analysis & Design**
   1. UML Diagram
   2. System Flow Structure
   3. Data Dictionary
   4. System Navigation
5. **Summary**
   1. Assumptions
   2. Limitation
   3. Conclusion
   4. Future Scope

**Chapter-1**

**Introduction**

**(1.1)**

**Introduction of Company :-** The Company Voculas Software Pvt. Ltd. is a Private company in Ahmedabad.

**Company Name : -** Voculus Software Pvt. Ltd.

**Owner of the company :-** Vishal Shah

**Website :-** www.voculus.in

**Address :-** 106, Apollo Arcade, Near R. C. Technical Road

Chhadlodiya Ahmedabad Ahmedabad-382481



**About the company**

* + - * Android app development
      * Website development
      * Computer software development
      * Windows mobile app development
* Transforming your vision into realitywith the motto of ‘Transforming Your Vision into Reality’, we have developed Android apps, iPhone apps and Web designing. We offer you with a wide range of services with our mastery in the development of customizedmobile apps. Our approach towards web developing and providing software solutions to the companies across the globe provides us with the niche over the other companies.
* Help you achieve global success and expand your business we understand your business objectives and develop mobile applications and business solutions tailored to brilliance. Providing customized solutions to the companies ranging from all kinds of Android apps to iPhone apps is the forte of the voculus. We help you transform your business by providing you with a magnificent websites that will serve your customers and provide them with the best experience. We provide you with social media integration that will help you achieve global success and expand your business.
* Satisfied customers across the globe we provide you with an ultimate experience of understanding your ideas and optimizing them with better solutions. With providing services in India, Canada, Italy, Brazil, Germany, Philippines we tend to provide our services of web developing and development of mobile apps without any time and place barriers. Our technique and efficiency to handle the problems has won us the appreciation of the satisfied customers across the globe.

**(1.2)**

**Project Detail**

In today’s technical world all the processes are done online. It reduces the time span of the work. People can surf any website according to their needs on single click. People can get instant reply, various and better options. They can compare things from different websites. In this large world things are getting much easier through internet. Things like booking tickets, shopping, etc. are mostly done online through websites like bookmyshow, flickstime, amazon, flipkart and many more.

**(1.2.1)**

**Current System**

Currently company does not have system to book online tickets. They book the tickets manually. They advertise their events in newspaper or Media per client requirement.

**(1.2.2)**

**Propose System**

Our System will provide a platform for the users who want to create their events easily and they can also buy various tickets for the events and user also buy promotional offers through our system.

**(1.3)**

**Scope**

Book online ticket

* Get discount through promotional coupon
* Use system wallet
* Easily get ticket
* Create own event
* Online payment
* Reminder on booked event
* Search category wise event
* Show past events
* Show upcoming events
* Manage own event

**(1.4)**

**Objective**

Our System provides the Interface to our client who wants to showcase their talent but lack the management skills to do so. By registering to our **Website**. The clients can create his/her own events and thus sell the tickets online. Our system helps Users to buy more tickets by giving promotional offers, cash back and discounts. Fore.g. A person Mr. X is living in Delhi who has very good knowledge in php language and then he moves to Ahmedabad but here he has no such platform to teach php. By joining in our **Website**, Event Admin can create his own event and can sell the tickets online and can earn money.

* **Admin Module**
* Super admin of the website is the head and he canmanage(add, update, delete)the users and he can also manage(add, update, delete) all events.
* **Event Module**
  + - various type of event
    - client create event
    - sell ticket for the event
    - sell promotional coupon
    - client can buy event ticket for that event
* **User Module**

All users of the website and application can,

* choose their favourite event of their choice
* Book ticket for that event
* give feedback on attended programmes
* Buy promotional offers
* Online payment

**Chapter 2**

**Requirement gathering**

**(2.1)**

**Objective of stakeholders**

There are 3 stakeholders for our website:

* Admin
* Event Admin
* Users
* **Admin**
* He is the owner of the website who has all the rights regarding to the website.
* The admin of the website can manage the members or events of the website.
* **Event Admin**
* Event Admin is the Admin who has only access to his particular event and can Add, Update and Delete for that event that he has created.
* Event Admin has all the rights regarding to the event that is been created by him and can give various types of discounts too.
* **Users**
* Users are the common Users who want to book the tickets and doing bookings online.
* Users are the person that buys the tickets and doing payment.

**(2.2)**

**Requirement gathering techniques: -**

**List of Techniques: -**

1. **Interview**
2. **Questionnaire**
3. **One-on-one Interview**
4. **Joint application development (JAD)**
5. **Prototyping**
6. **Following people around**
7. **Request for proposals**

**Techniques used: -**

**Interview: -**

Since there is only one Admin for the system, we are going for interview.

* You want to delete an event? If you want to delete an event then why?
* You want logical delete or physical delete?
* You need any specific requirements in website?
* Which method you like to confirmation? Text message or Email?
* How you want to search any event in admin module?
* Which method would you like to get reminder of your event day? Text message or Email?

**Questionnaire: -**

Since you are supposed to cover large number of users, we are going for interview.

* Which method you want to use for select city and state?
* Which specific requirements you want to in event description?
* What type of promotional offers you are interesting?
* Which method you prefer to do online payment?
* Which type of event you want to attend?
* You want to buy additional coupon for discount?
* Which method is comfortable to contact Event admin?

**Which technique is applicable for which domain / stakeholder?**

Interview is suitable for Event Admin.

Questionnaire is suitable for the users or buyers.

**Why is it suitable for your domain?**

Interview is suitable for Event Admin because Admin is the only one person who access admin module and we need to take requirements from only one person. Interview is easy for less number of people. We can get specific answers in interview and we understand admin requirements properly.

Questionnaire is used to collect large amount of information from a large number of information from a large number of people in a short period of time and in a relatively cost effective way. That’s why we used questionnaires for user module.

**(2.3)**

**List of Requirements: -**

Our Website has no such kind of special requirements to run and operate this website and functionality.

* User wants dropdown list for to select the city that it should be easy to choose.
* Payment method should be online. It will be comfortable for User and Event Admin as well.
* Online payment method should have all the kinds of payment gathering like net-banking, debit card, credit card, paytm etc.
* User wants search filter in the website with a view to search easily and efficiently.
* There should be search engine for the User to search for the events easily.
* There should be different types of categories in the events to access particular kind of events.
* If User cancels his ticket the amount he paid should be returned to his app-wallet.

**(2.4)**

**Project Definition: -**

We are developing a system where users can book and buy Tickets online where One wants to share his opinions and views but he has no proper environment or any platform, then our website helps to them that they can book their events online and buyers can buy the tickets of that particular event. Users can also get various types of discounts, cash back and promotional offers. So, our **Website**and **Android Application**creates a platform for the buyers and sellers.

# **3. Project management and planning**

## **(3.1)**

## **Feasibility study**

The main aim of the feasibility study is to determine that it would be financially and technically feasible to develop the product or not. The purpose of feasibility study is not to solve the problem, but to determine whether the problem is worth solving.

**(3.1.1)**

**Technical feasibility**

The technical feasibility study compares the level of technology available in the software development firm and the level of technology required for the development of the product. Here the level of technology consists of the programming language, the hardware resources, other software tools, etc.

In this system technical feasibility was ensured on in the following factors:

* Availability of servers and open-source PHP.
* It accesses software engineering work products in an effort to remove errors before they are propagated to next activity.
* One of the most effective quality assurance mechanisms can be applied from the inception of a project with this.
* The hierarchy of technical work within the software process is activities, encompassing actions, populated by tasks.

### **(3.1.2)**

### **Operational feasibility**

Operational feasibility study tests the operational scope of the software to be developed. It is checked that if the system is actually can be useful when implemented.

Our system is operationally feasible in the following ways:

1. As each class is developed, the team develops a unit test to exercise each operation according to its specified personality.
2. An effective catalyst for customer feedback is a portion of an operational system.
3. If a difficult design problem is encountered as part of the design of a story, XP-recommends the immediate creation of an operational prototype of that portion of the design.

### **(3.1.3)**

### **Economical feasibility**

The economic feasibility study evaluates the cost of the software development against the ultimate income or benefits gets from the developed system. There must be scope for profit after the successful completion of the project.

The system is economical because:

1. In economic feasibility, the most important is [cost-benefit analysis](http://www.freetutes.com/systemanalysis/sa3-cost-benefit-analysis.html).
2. Additional investment is needed in developing tools because of open-source PHP platform.
3. Due to economic feasibility we can estimate and make changes according to the budget.

## **(3.2)**

## **Hardware and Software Requirement**

Software Requirement:

* Windows 7 and higher and/or equivalent operating system.
* Java Script enabled browser.
* Browsers: Safari, IE 7, Edge, Chrome,Mozilla

Hardware Requirement:

* Intel P4 and higher and/or equivalent processor system.
* 1 GB RAM.
* 60 GB HDD.

## **(3.3)**

## **Project Planning**

**(3.3.1) Project breakdown structure**

**(3.3.2)**

**Gantt Chart**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Activities** | **July** | **Aug** | **Sep** | **Oct** | **Nov** |  |
| Project Scope |  |  |  |  |  |  |
| Research |  |  |  |  |  |  |
| Requirement Gathering |  |  |  |  |  |  |
| Analysis |  |  |  |  |  |  |
| Designing |  |  |  |  |  |  |

## **(3.4)**

## **Process Model**

## As new requirements can arise in future incremental models are used. With the help of it we can fulfil maximum user requirements.In every increment the needs of the client are kept in mind and more features and functions are added. These increments form a base for customer evaluation. Many features can be added after the development of the system that serves the main purpose.If there are less number of employees to work on the project Incremental development model is very useful to complete the project before the deadline.

***Incremental Model***

* This model is more flexible – less costly to change scope and requirements.
* It is easier to test and debug during a smaller iteration.
* In this model customer can respond to each built.
* Lowers initial delivery cost.
* This process model helps to manage technical risks.
* Initial product delivery is faster.

# Incremental Model

# **4. Diagram**

**(4.1) UML Diagram**

## **(4.1.1)**

## **Use Case**

**(Admin)**



**(User)**

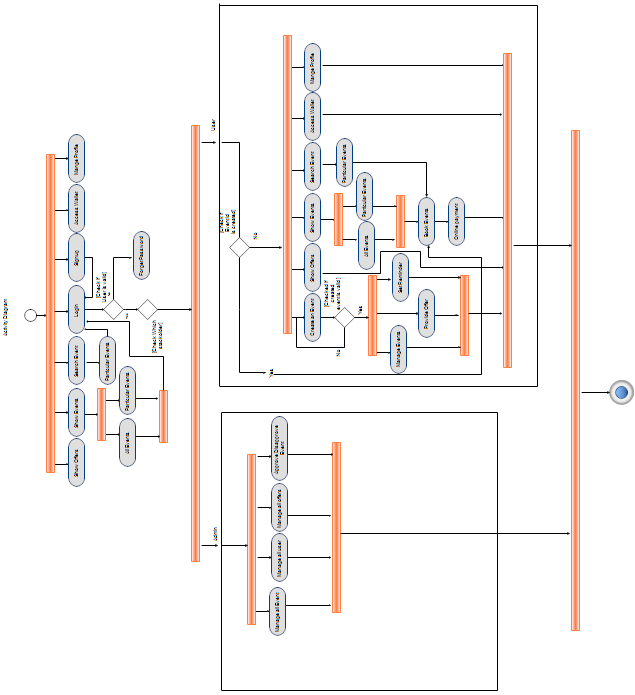


**(Event Admin)**



## **(4.1.2)**

## **Activity Diagram**



## **(4.1.3)**

## **Class Diagram**

## 

## **(4.1.4)**

## **State Diagram**

**(Admin)**



**(User)**



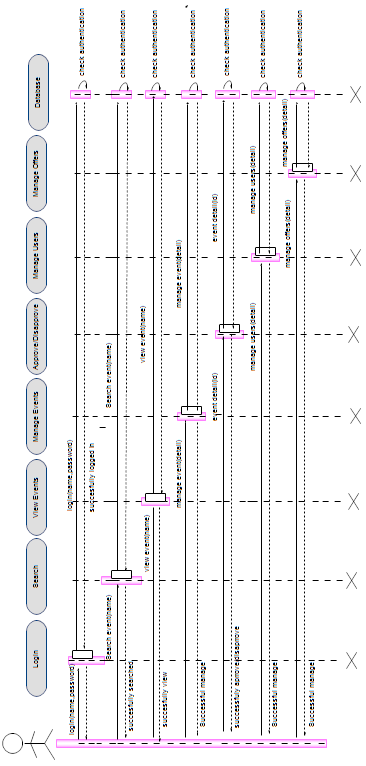
**(Event Admin)**



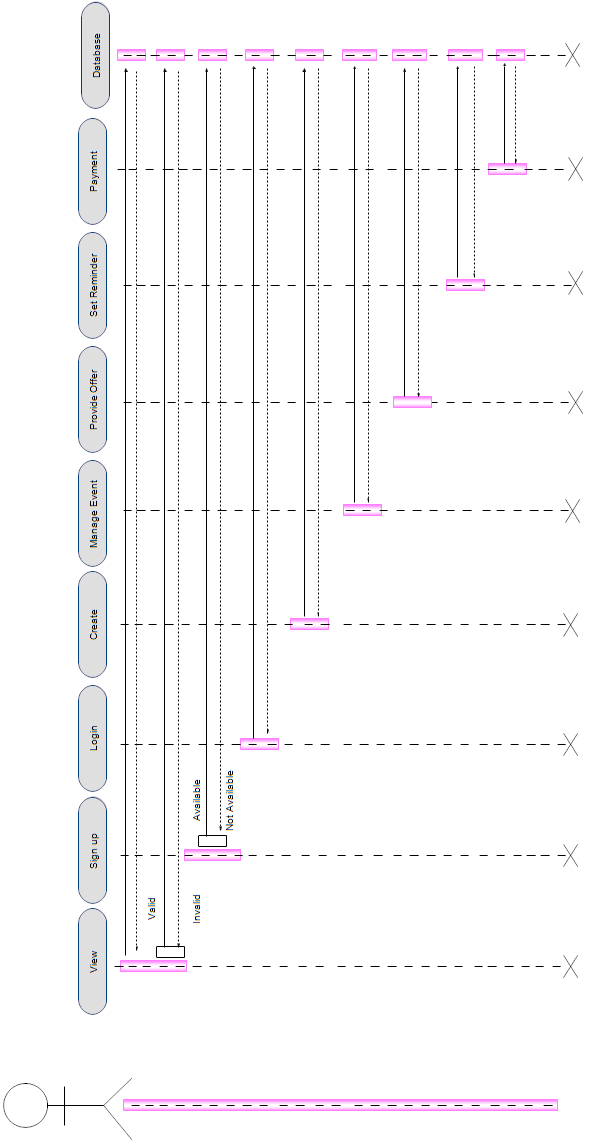
## **(4.1.5)**

## **Sequence Diagram**

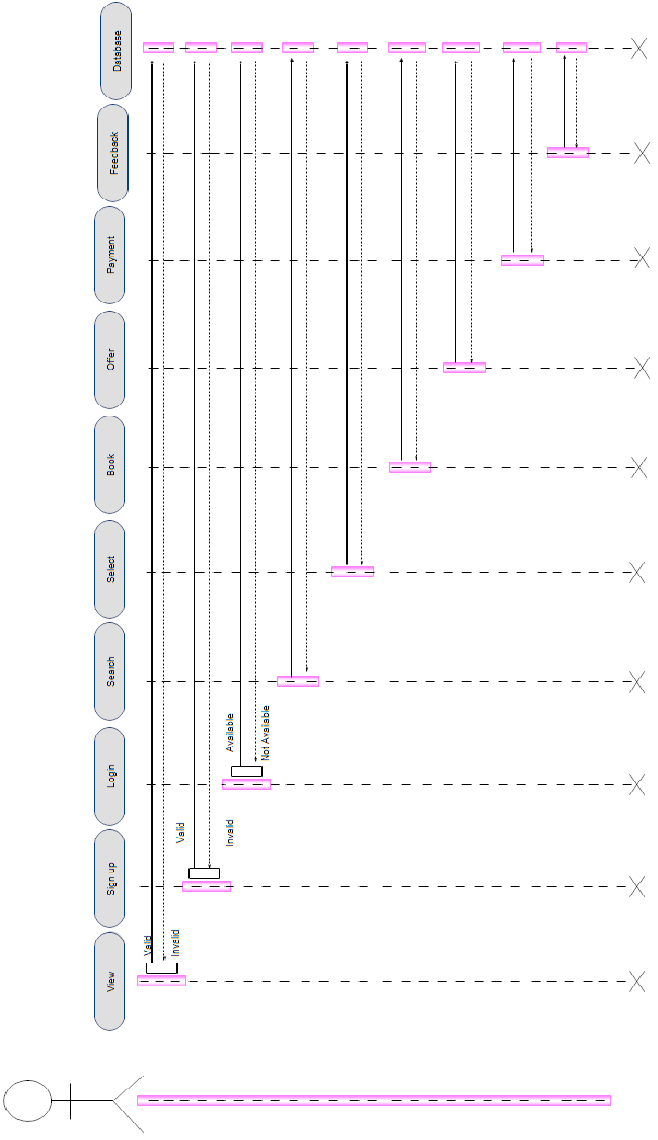
**(Admin)**



**(Event Admin)**



**(User)**



## **(4.1.6)**

## **Deployment Diagram**

**(4.2) Flowchart**

**(4.3) Data Dictionary**

User Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| pk\_email\_id | Varchar | 50 | Primary key | Store unique email id |
| User\_password | Varchar | 20 | Not null | Store password of the user |
| User\_mobile\_no | Varchar | 11 | Not null | Store mobile\_no |
| Fk\_city\_id | Varchar | 20 | Foreign key | Store user's City |
| User\_name | Varchar | 50 | Not null | Store User name |
| User\_type | Varchar | 20 | Not null | Store User type |

**References-** City Table

City Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_city\_id | Int |  | Primary key | Store unique city id |
| City\_name | Varchar | 20 | Not null | Store city name |

Event Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_event\_id | Int |  | Primary key | Store unique event id |
| Event\_name | Varchar | 50 | Not null | Store event name |
| Event\_logo | Varchar | 50 | Allow Null | Store event logo |
| Event\_slogan | Varchar | 50 | Allow Null | Store event slogan |
| Event\_des | Varchar | 200 | Not null | Store event description |
| Fk\_venue\_id | Int |  | Foreign key | Store unique venue id |
| Event\_date | date | 50 | Not null | Store event date |
| Event\_time | Varchar | 50 | Not null | Store event time |
|  |  |  |  |  |
| Event\_ticket | Int |  | Not null | Store event ticket number |
| Event\_price | Int |  | Not null | Store event price number |
| Fk\_cat\_id | Int |  | Foreign key | Store unique category id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Fk\_offer\_id | Int |  | Foreign key | Store unique offer id |
| Event\_cnt | Int |  | Allow Null | Store event sold ticket count number |

**References-** Venue table , Category table, User table, Offer table

Venue Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_venue\_id | Int |  | Primary key | Store unique venue id |
| Venue\_name | Varchar | 50 | Not null | Store venue name |
| Venue\_add | Varchar | 200 | Not null | Store venue address |

Category Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_cat\_id | Int |  | Primary key | Store unique category id |
| Cat\_name | Varchar | 50 | Not null | Store Category name |

Book Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_book\_id | Int |  | Primary key | Store unique ticket book id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Fk\_event\_id | Int |  | Foreign key | Store unique event id |
| Ticket\_cnt | Int |  | Not Null | Store ticket cnt |
| Tot\_ticket\_amnt | Int |  | Not Null | Store total ticket amount |

**References-** User table, Event table

Testimonial Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_review\_id | Int | 50 | Primary key | Store unique review id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Review\_des | Varchar | 200 | Not null | Store review description |
| Review\_like | Int |  | Allow Null | Store review like count |

**References-** User table

Event question Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_que\_tbl | Int |  | Primary key | Store unique question id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Fk\_event\_id | Int |  | Foreign key | Store unique event id |
| Que\_desc | Varchar | 200 | Not null | Store question description |
| Que\_date | Varchar | 500 | Not null | Store question date |

**References-** User table, Event table

Answer Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_ans\_id | Int |  | Primary key | Store unique answer id |
| Fk\_que\_id | Int |  | Foreign key | Store unique question id |
| Fk\_event\_id | Int |  | Foreign key | Store unique event id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Ans\_desc | Varchar | 200 | Not null | Store answer description |
| Ans\_date | Varchar | 50 | Not null | Store Answer date |

**References-** Event question table, User table

Wallet table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_wallet\_id | Int |  | Primary key | Store unique wallet id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Wallet\_amnt | Int |  | Allow Null | Store wallet amount |
| Wallet\_date | Varchar | 50 | Allow Null | Store wallet date |

**References-** User table

Offer table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_offer\_id | Int |  | Primary key | Store Unique offer id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Offer\_price | Varchar | 50 | Not null | Store offer price |
| Offer\_promocode | Varchar | 200 | Not null | Store offer promocode |
| Offer\_photo | Varchar | 200 | Allow Null | Store offer photo |
| Fk\_event\_id | Int |  | Foreign key | Store unique event id |

**References-** User table, Event table

Like Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| Pk\_like\_id | Int |  | Primary key | Store unique event like id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Fk\_event\_id | Int |  | Foreign key | Store unique event id |
| Like | Boolean |  | Allow null | Store like count |
| Dislike | Boolean |  | Allow null | Store dislike count |

**References-** User table, Event table

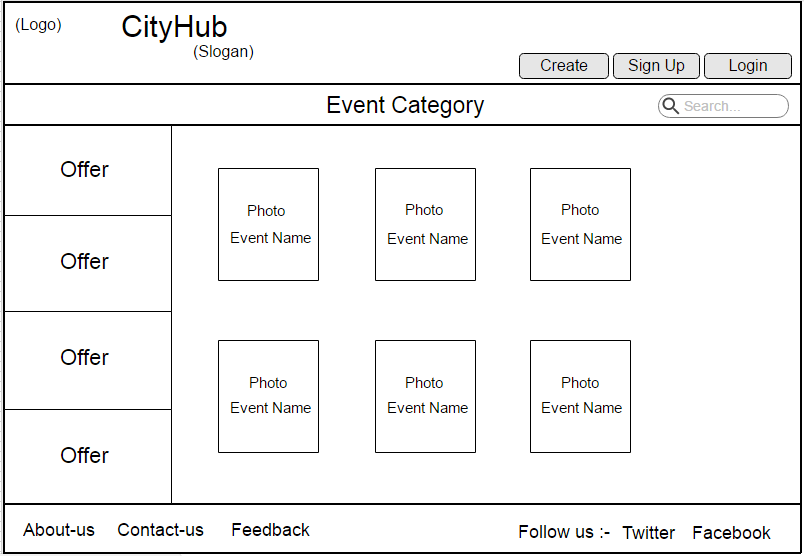
Payment table

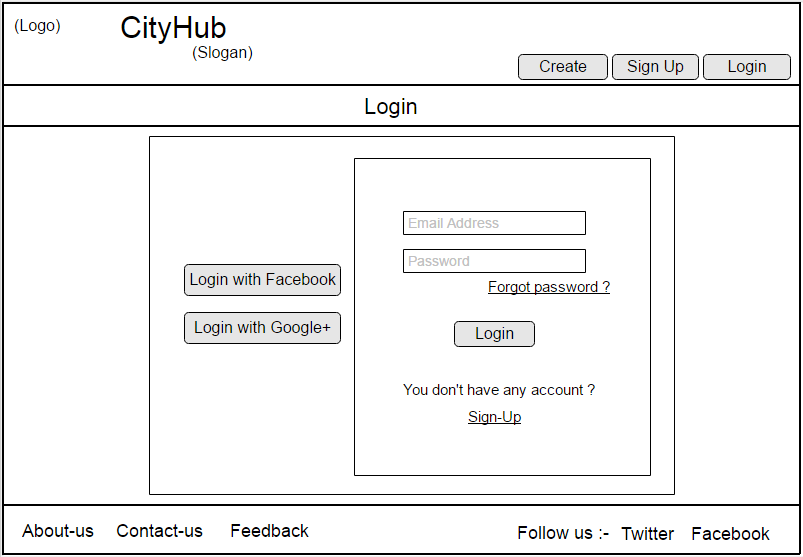
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Datatype | Size | Constraint | Description |
| pk\_payment\_id | Int |  | Primary key | Store unique payment id |
| Fk\_email\_id | Varchar | 50 | Foreign key | Store unique email id |
| Fk\_event\_id | Int |  | Foreign key | Store unique event id |
| payment\_method | Boolean |  | Allow null | Store payment method |
| payment\_date | Boolean |  | Allow null | Store payment date |

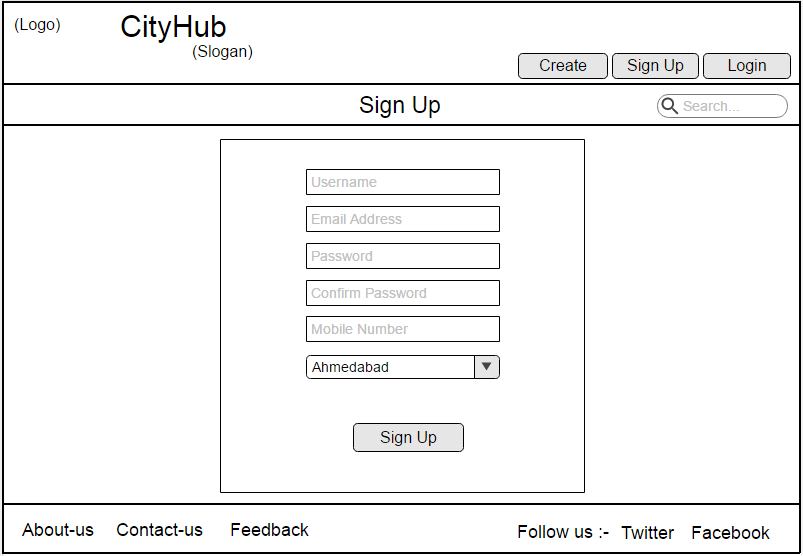
**References-** User table, Event table

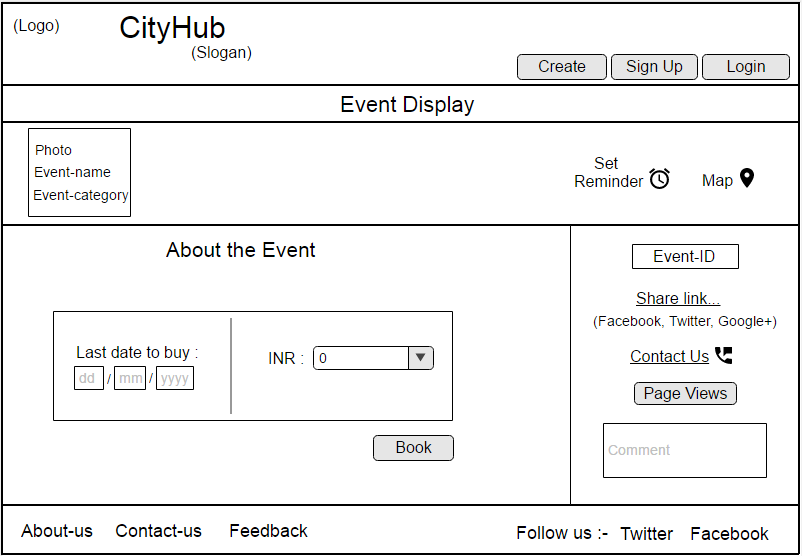
**(4.4)**

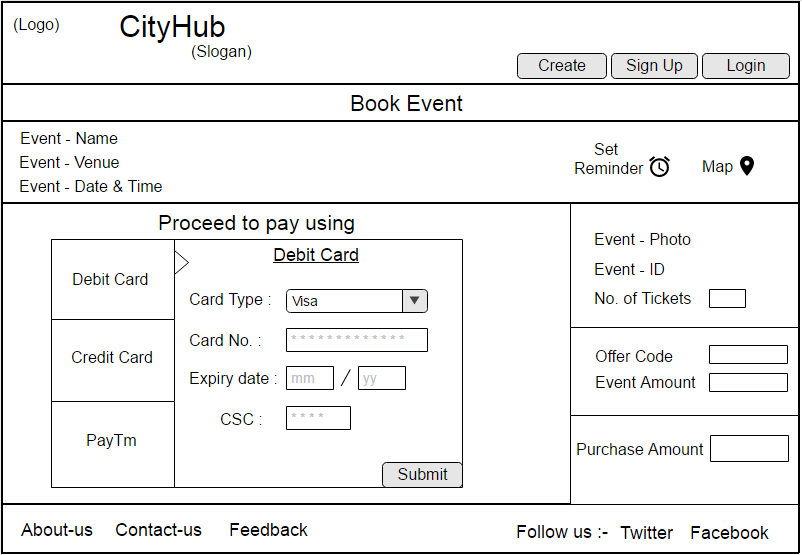
U**ser Interface (Wire Frames)**

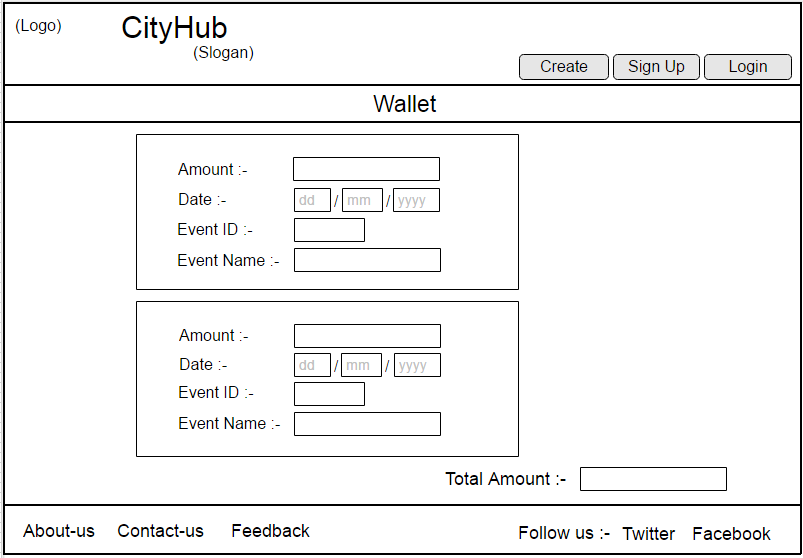
****

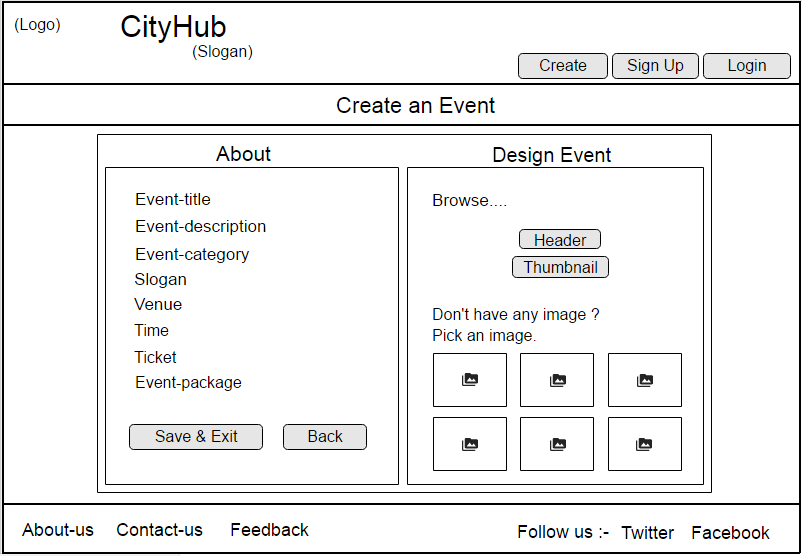
****

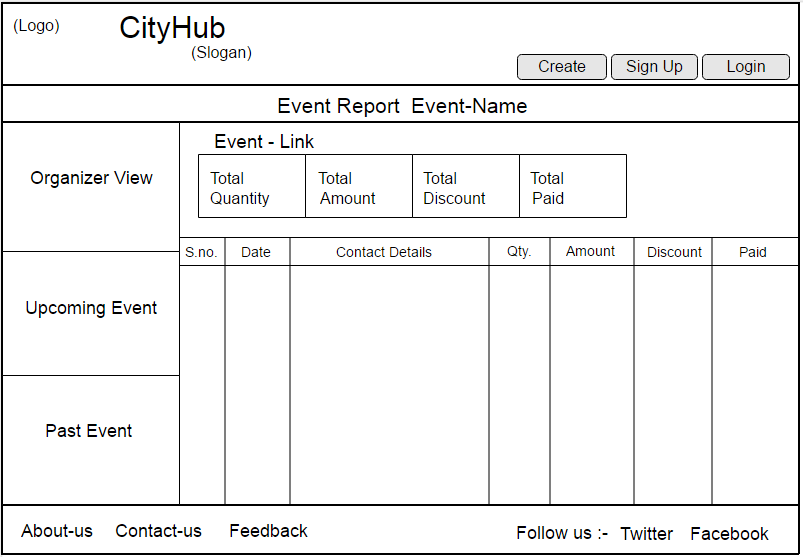
****

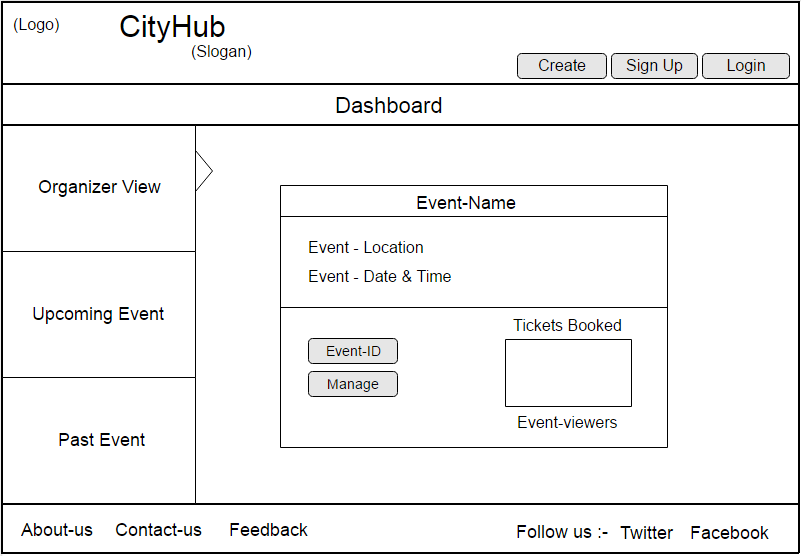
****

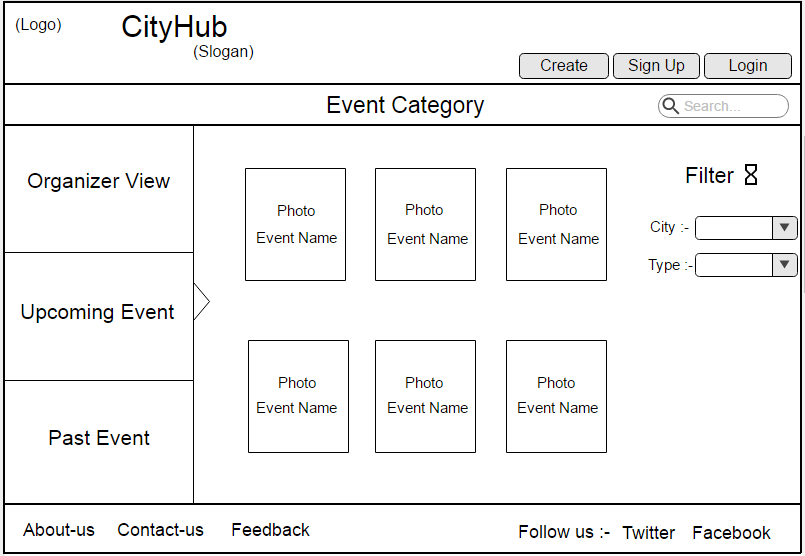
****

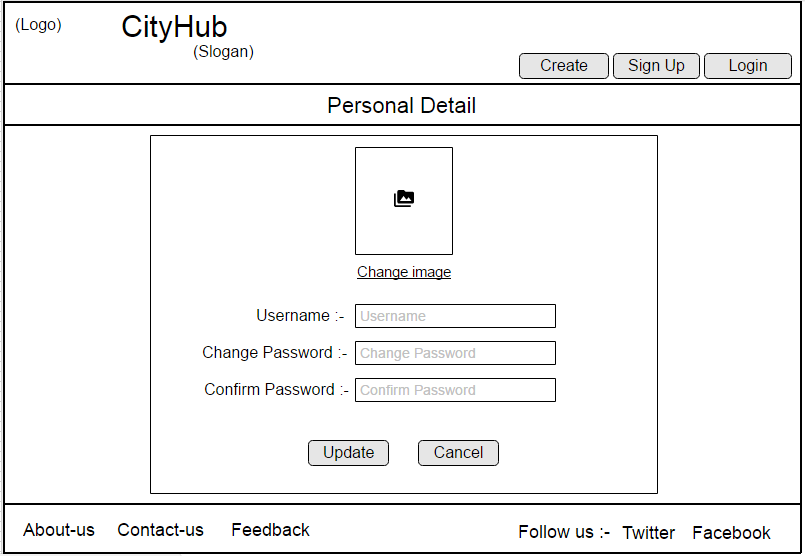
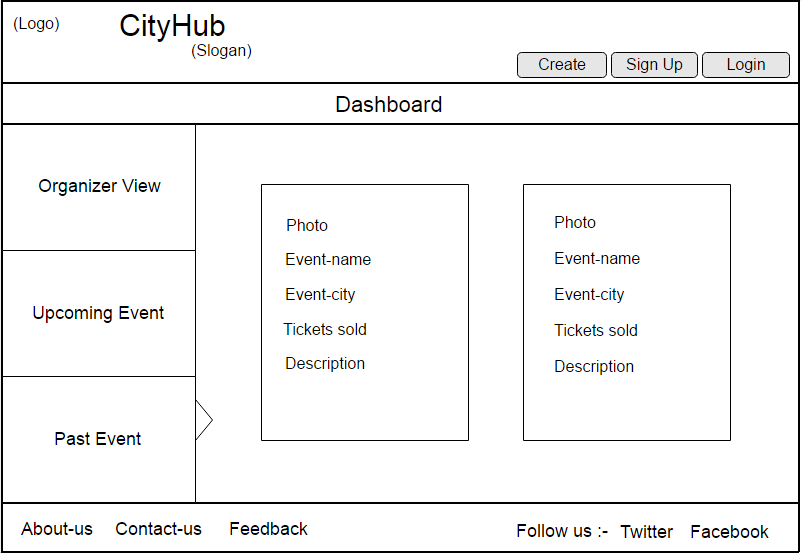
****

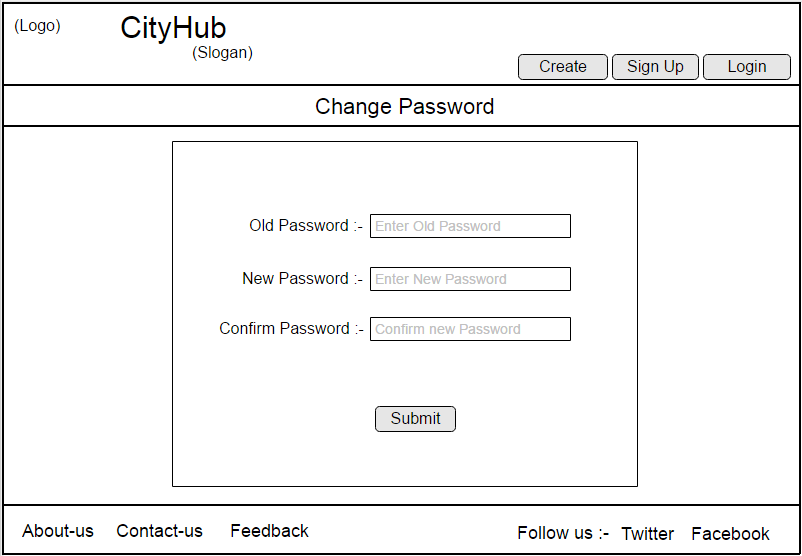
****

****

****

****

****

****

**(4.5) System Navigation (Road Map)**