# Design Specification

Project design provide all detail of system development and key feature, structure, schedules, task are planned out. As well as components, models, interfaces and data. That information involved directly or indirectly in system development. In this project we create the blue print. Design project divided into three model these models are structural, Behavior and database model.

In Bus ticket booking system we use star UML tool to use create class diagram, DFD diagram, Activity diagram, sequence diagram. In this project I have use unified modelling language (UML).

# Structural design:

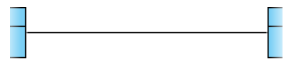
Structural model is the view of a system that emphasizes the structure of the objects, including their classifiers, relationships attributes and operations. The structural design helps through managing the classes and object to form the large structure. They help to manage and identify the larger system and relation to identify in the simple understanding large figure into simple one. By the help of braking system into smaller unit.

# Class diagram

Class diagram is a static model which is one of the most useful types of diagram in UML. in this diagram clarifies all classes in the system, classes, attributes, operation and relations between objects.

Notations of class Diagram

Association:



This association generally capable of showing the real problems of domain between the two classes.

Inheritances:



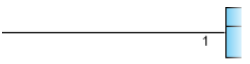
It represents the simple meaning that the contains in the head class will have same features in the child class or sub class.

Realization:



Relationship between the blue print of classes with the object related in respective details.

Exactly one:



So, the one entity has only one relationship with other class.

Zero or one



It represents that the single entity will be related with each unit other in relationship.

Zero or more



The entity has relationship can be with more entity in single class.

One or more



Each single entity will have relations with many entities in other class.

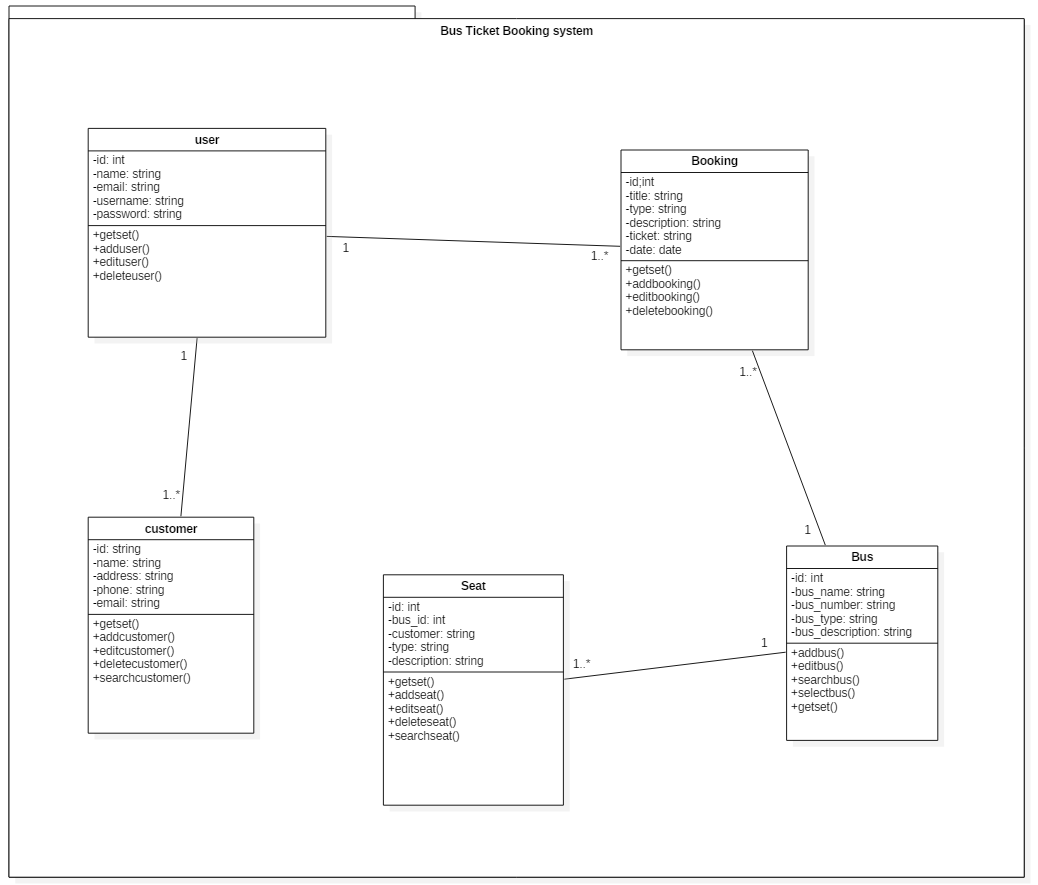
**N**

Figure 1: class diagram

The class diagram helps to identify the total number of classes in the current project and their relation related with. Makes easy to understand the basic scenario from the diagram. Identify the entity with the attribute and as well method used by each class.

# Data flow diagram

Data flow diagram is the graphical representation of flow of data from one component to another component in any information system.

Symbol in DFD are given below:

Process:

The activates carried out by the system which use and transform information. Beginning of the system where actual process began.

Data flow:

The data flow is noted as names the arrow. Flows between the entity to process and process to the databases. This symbol is used to determine the flow of process.

External Entities:

The source from which information flow in to the system and recipient of information leaving the system.

Data Store:

where information store in the system. This symbol represents the flow of data from different process to database.

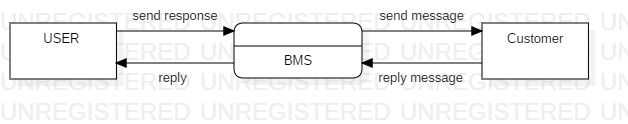


Figure 2: 0 level Data flow diagram

This is the first stage of the DFD where it represents the whole system between the user, customer and management system in a simple form. This can be the simple layer of data flow diagram of level 1.

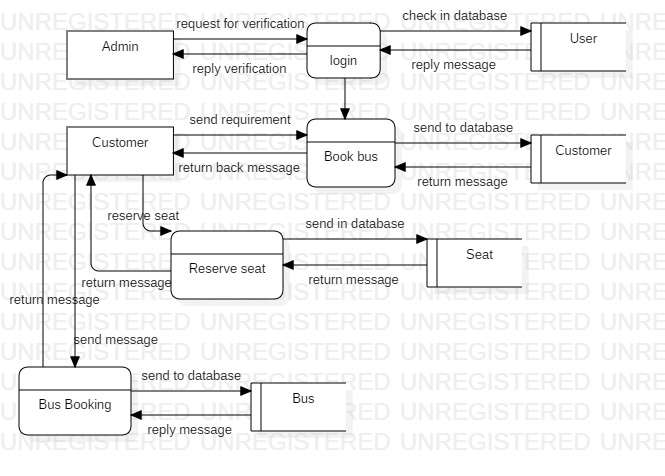


Figure 3: one level data flow diagram

The level1 DFD shows the advance form of diagram of flow of data than 0level DFD. Because it represents the better flow of data how actually dataflows from the entity to the process and process to the database. Middle interaction can be visible from this diagram.

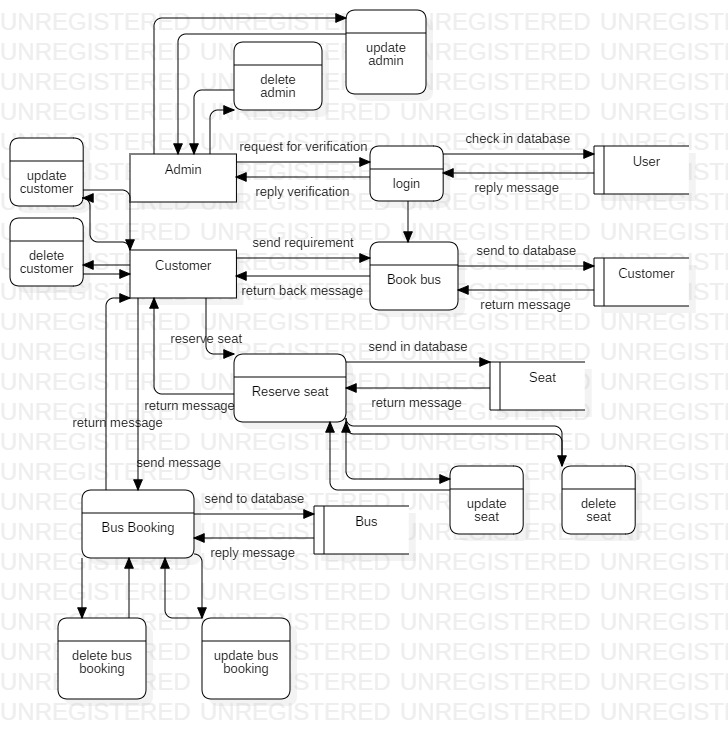


Figure 4: two level data flow diagrams

This level of DFD shows the all the information of flows of information and their different features between them. How the single data moves around between the entity and process before going to the databases. So, number of activity of flows of data can be identify in this DFD.

# Behavioral Design:

Behavioral is also knowing as Dynamic modelling in this design system which deals with every types of behavior in this system. In this model we have involved activity and Sequence diagram in this Bus Ticket booking system.

# Activity Diagram

It is a special kind of a state chart diagram that shows the flow from activity to activity within system. Activity diagram addresses the dynamic of system. They are especially important in modelling the function of system and emphasis the flow of control among objects. And activity diagram defines different part of the system.

Symbol of Activity diagram are given below



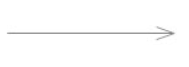
Start point:

This symbol is the beginning of the activity. The start points noted by black circle.



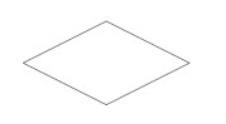
Activity:

this symbol is the main block of activity. Shows the flows of information between activity.



Action Flow:

Shows the way where this information flows from where to where.



Decision symbol:

Decision symbol represented by a diamond. In this symbol single input into two or more outputs.



End point symbol:

End point symbol is the last point of activity. This symbol represented by outlined black circle.

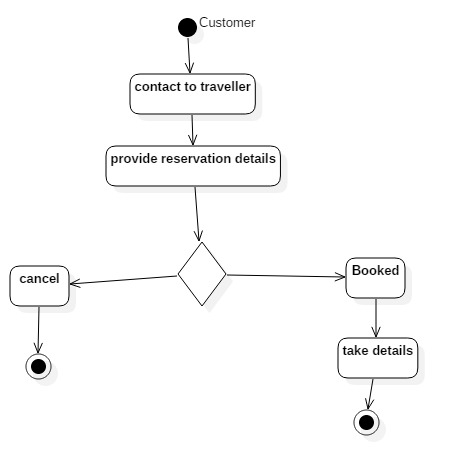


Figure 5: first Activity Diagram

The activity of customer shown in above diagram where the activity of customer shown interaction with system when they interact with user. This can only show the single activities of the customer over the whole activities.

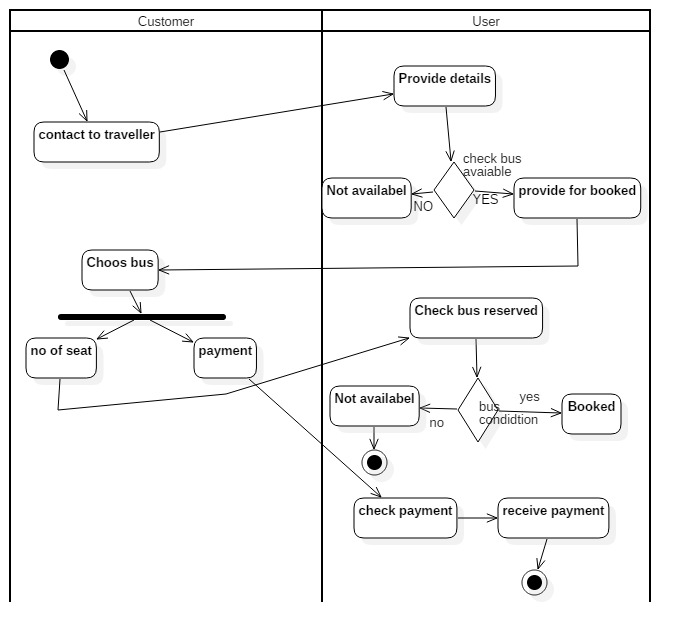


Figure 6: Activity Diagram

This diagram is the complete diagram between the user and the customer shows their activities between them. How the system begins and where the information flows from the customer to the system us and again goes to the database to record keeping purpose. Different symbols represent the flow of data between the system use and the customer.

# Sequence Diagram

A Sequence diagram is an interaction diagram that emphasizes the time ordering the message. Sequence diagram is important in UML (unified modelling language) that displays the transactions of message between lifelines.

Lifeline: it determines the individual contributor in the interaction. Also known as lifeline of the single entity over the whole system. It shows the activity of active user that involves during the whole system. How much lifeline needed to each entity at whole process with system.

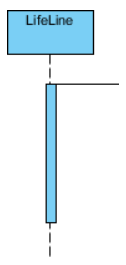




Class Roles: represent the object involves in the diagram. This shows that the which object involves during the flow of information. Because the information can flow from the object to object.

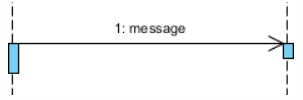
Activation:

The symbols represent the active of the object over the lifeline. The activation increase as the user involvement is more. More the interaction more long will be the active.



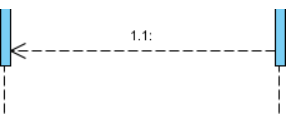
Call message:

call message defines the particularly communicate between lifeline of an interaction. Send messages from the different activation to the another just show what messages flows between them.



Return message:

When there is flow of message from one user to another there will be the return of responses massage. So, this symbol is used to return the message from object.



Self-Message:

self-message defines calling of message in same life. When there is message between same entity th same/ self-message icon is used between single unit.



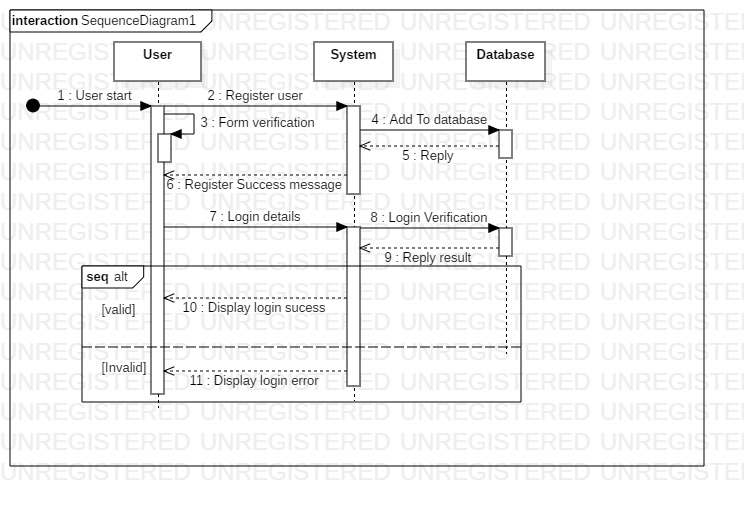


Figure 7: one level Sequence Diagram

This is the beginning of the sequence diagram of this assignment where the flow of information shown in the figure. The different flow between the system interaction are represent. Shows the flow of data and responses back message between the lifeline entity with system and database.

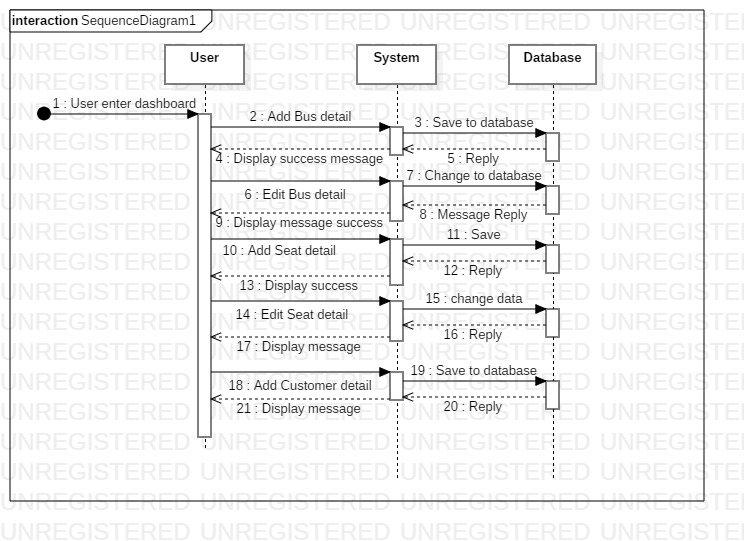


Figure 8: Two level Sequence diagram

The second stages of the sequence diagram from where the user enter into the dashboard after login into system successfully. And analysis the all the activities done by the user with their activities and responses from them.

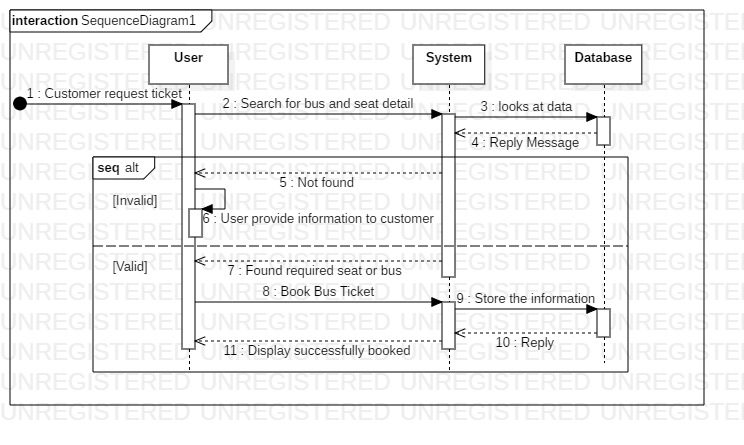


Figure 9: Three level sequence Diagram

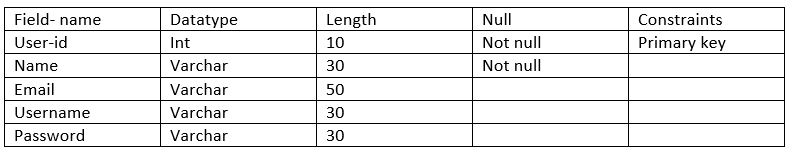
This is the part of remaining of sequence diagram of above and parts of search located here in the above diagram.

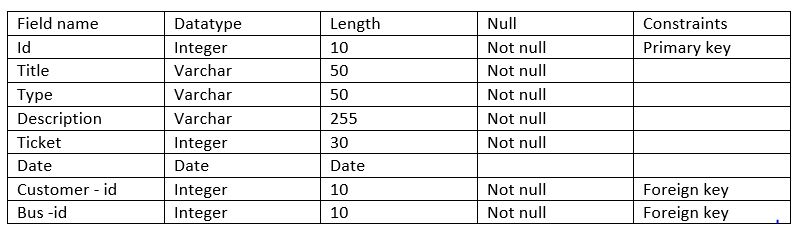
# Database Model

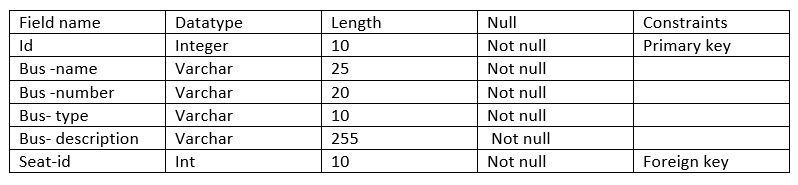
Database model is the logical structure of database model and database model is the type of data model. It is the relational model and database model is the table-based format. In this model we can store the data for Bus ticket booking system.

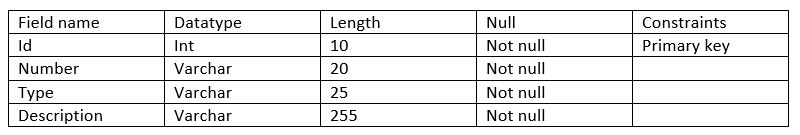
# Data dictionary

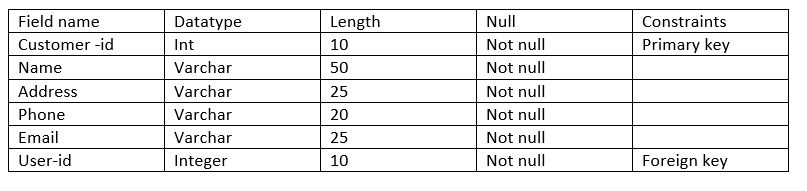
Data dictionary is files or set of files. it records all the objects in the database. It is also dictionary of data and store all information in database. Each table in the database and each unit have their own data dictionary. This data dictionary consists of different hidden features or metadata of data. Some of them are include in my project are sown below:











# Entity Relation diagram

Entity Relation diagram is the logical structure and it shows relationship among entities sets. In this diagram the main component is entities set and relationship set.

Bus ticket booking system is given below:

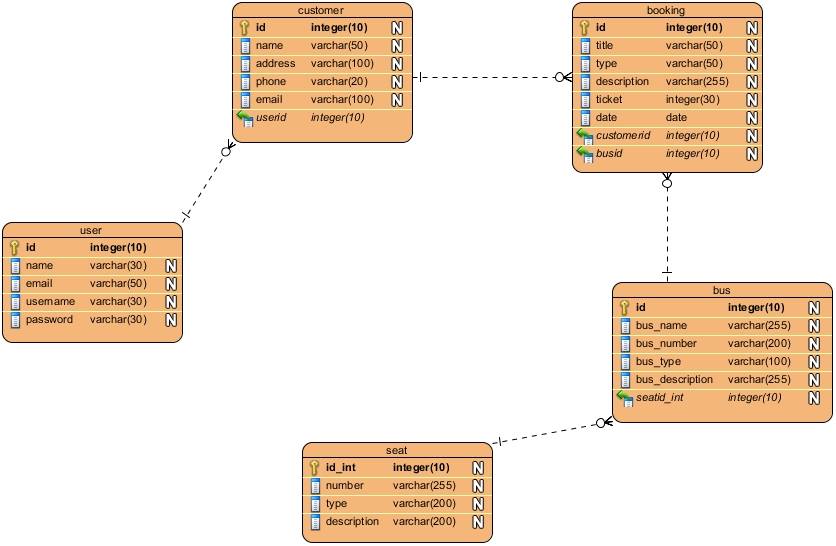


Figure 10: Entity relation Diagram

This above diagram shows the relationship between the different classes that involves in the project. Shows the kinds of relationship between them how they were interrelated with each other. The role of relationship affects the foreign key between them. The ER-diagram helps to simplify the relationship that how they inter-related with each other in system. Simplify the work to the programmer as well as fr those who need to understand it.

# System Architecture

System Architecture is the conceptual model that defines the structure, behavior and more views of stem. I have used 2 tier system Architecture.

Our system is based on the two tiers because the data is store on the same device and only between application and database.

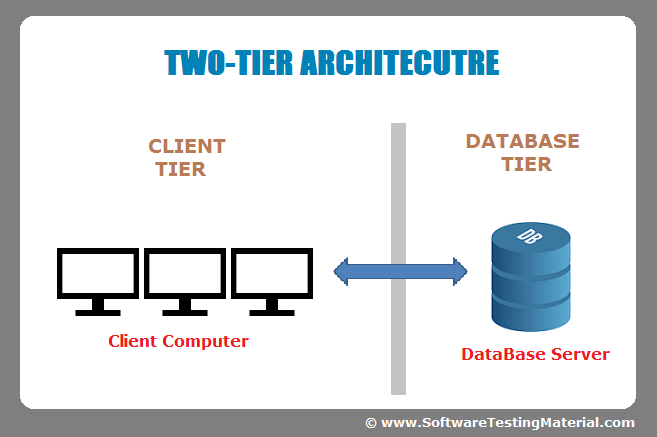


Figure 11: two Tier System Architecture

Generally, these kinds of architecture involve in the system where the data is accessible form database that are in the same level. In my condition project the two-tier architecture is preferred as the major because the server does not need to place from away. The system used local storage to store the data.

The client tire:

This represent the users contributing from the system. Generally, consist as the application or interface from that control the overall activities into the database tire. Client tire can only operate from the user interface. The user access, display data from this tire only using the interface and limited available of data.

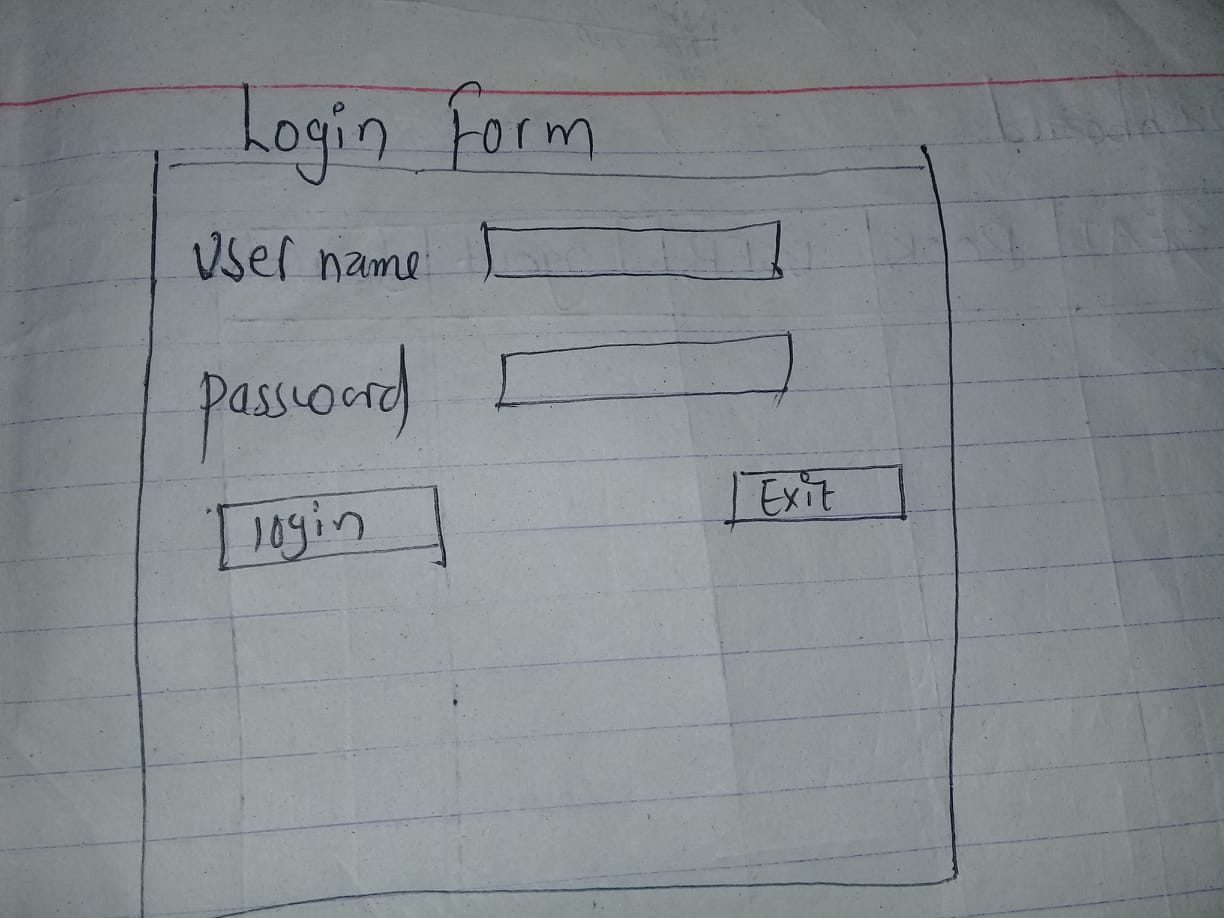
The database tire:

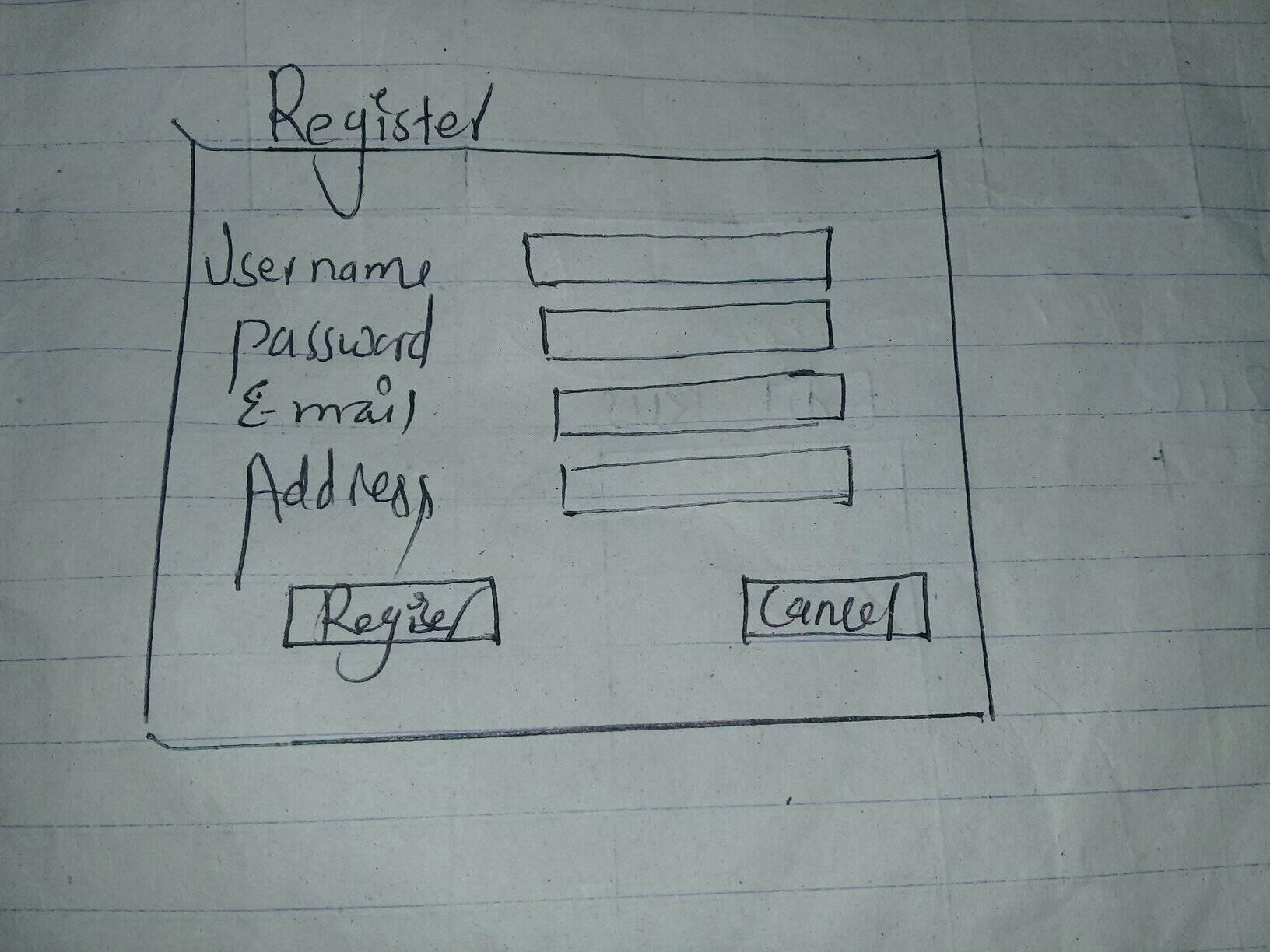
This tire includes of all the data that comes out from the user tires. All kinds of system changes can be implemented in this tire. But the user tire has not been completely access into this tire. Because of sensitive data only the approved user has access into this system.

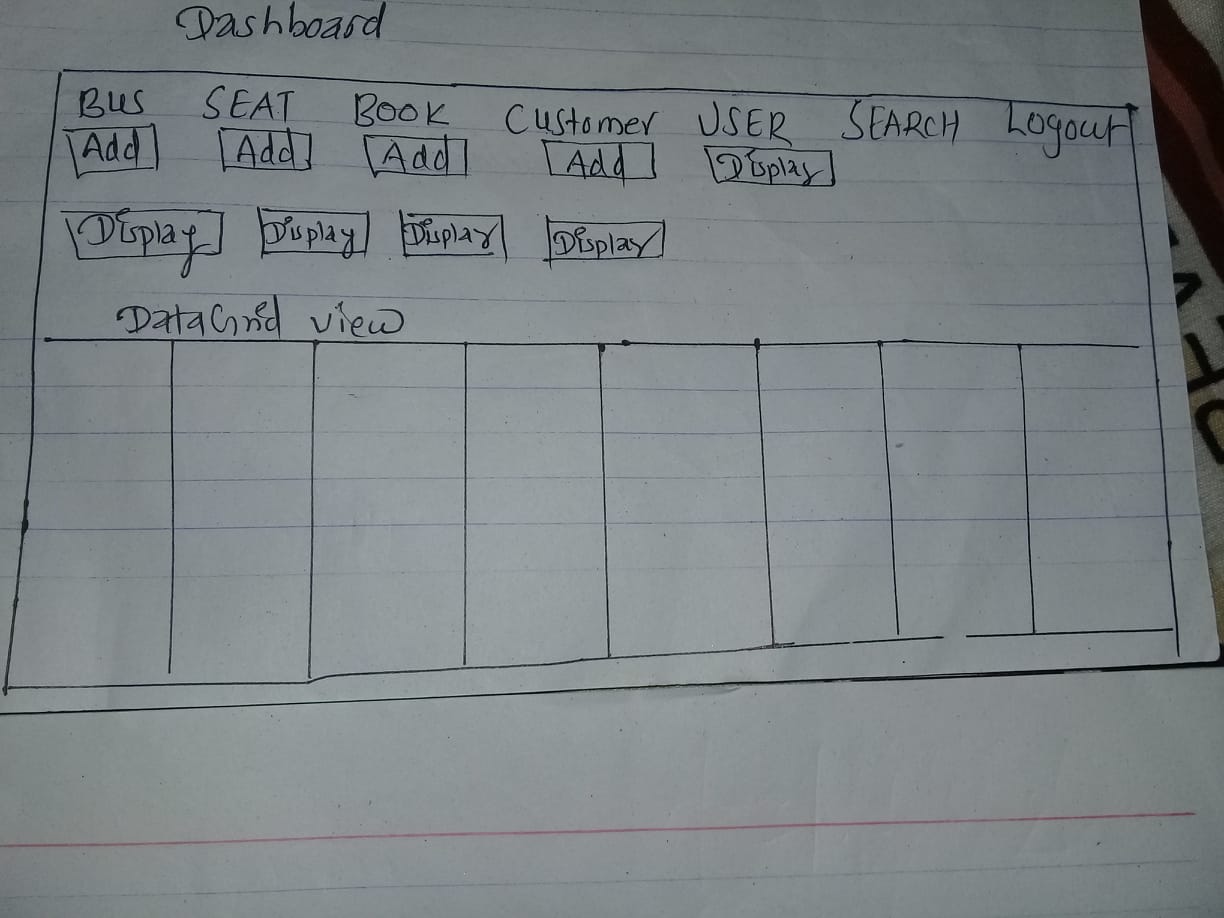
# Prototyping

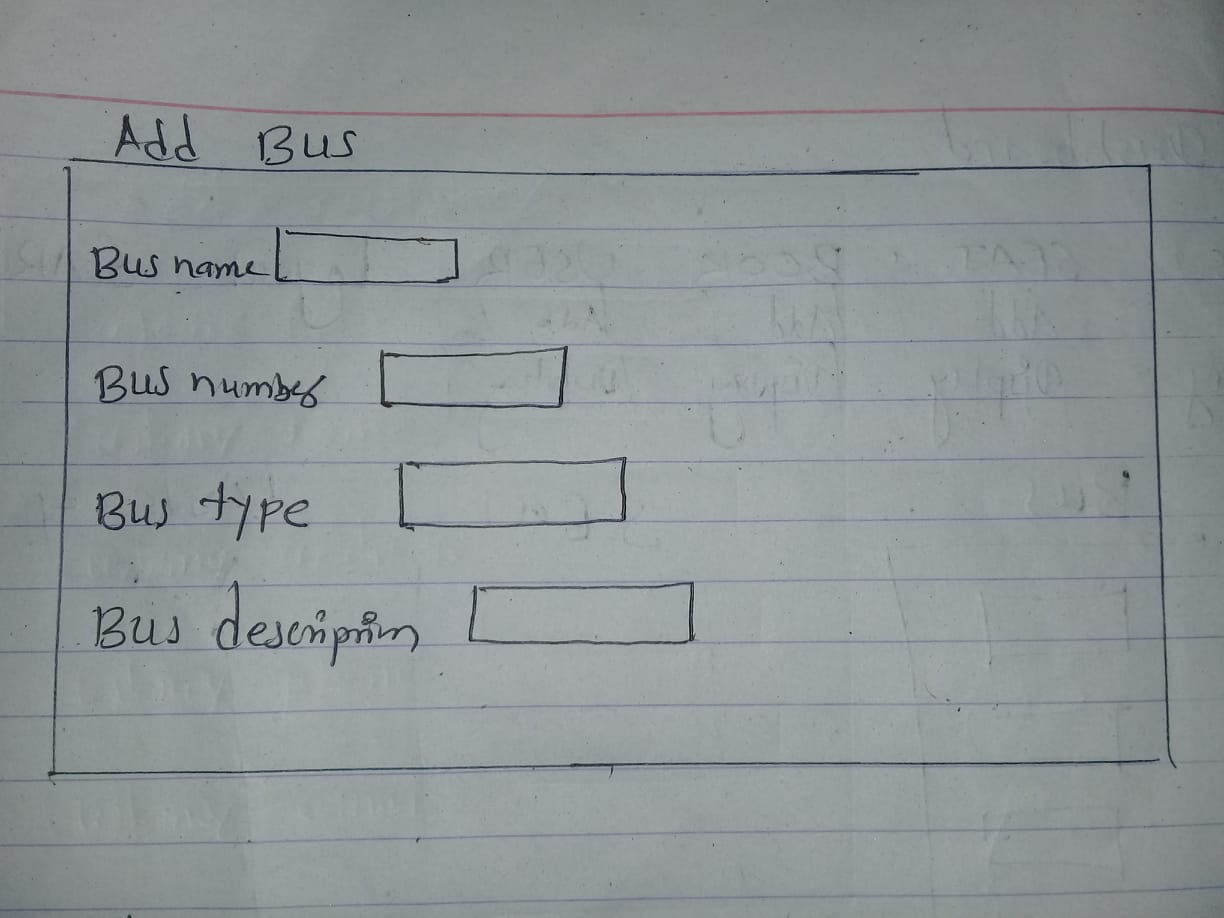
A prototype is working model that does not normally have all the required features or provide all the functionality of the final system. The prototype can help to identify the basis standard of the system. The requirement of all the things can be identify easily in the prototype. This helps to client to understand how their system looks in the future and what kinds of change does it requires to change before it implemented.

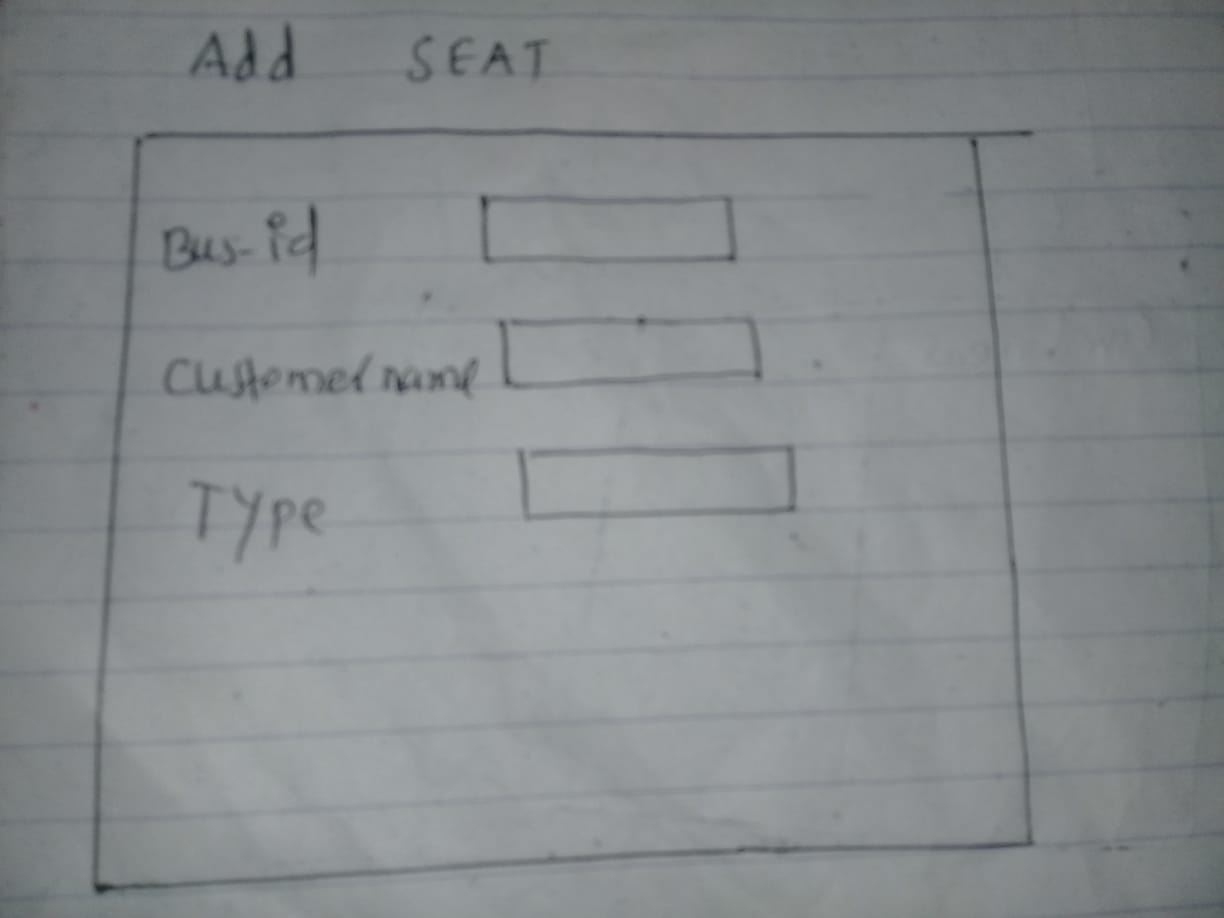
Prototype is the generally used when the requirement is unclear. It is used when the customer is unclear about the details the input, process and the output need of the software. This some of the prototypes of my projects are shown below:

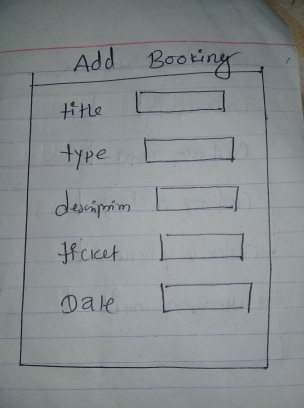


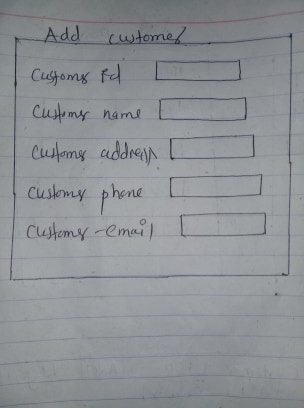


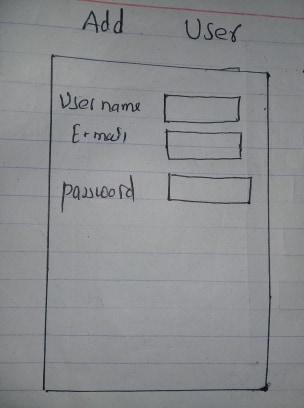












Conclusion

Finally, performed all the design of Bus ticket booking system. In this system I used different diagram includes structural, behavioral and database model. In structural model shows structure of class diagram and data flow diagram. In behavioral show the activity diagram and sequence diagram. Activity diagram shows the logical operation and sequence diagram shows the time and objects. In database model, we have generated data dictionary and Entity relationship diagram. At the last we make a prototype.