# 2. Analysis

## 2.1 Introduction

When we are building any software, there are certain steps that must be done and the process is known as the SDLC of Software Development Life Cycle. According to the principle of the software development life cycle or SDLC there are five sages in the life cycle. The first stage of the development of the software is Analysis. Analysis can be defined as a process of identifying, studying and managing of aims and various objectives of the software that is going to be built. We mainly analyze the certain software in order to find the aim and the objectives of it. This will also help to solve the problem and issue that could occur while we are developing the software which would help to ensure that all the parts and the components of the system is being worked efficiently and effectively. Analysis will also help to identify the requirement of the system that should be need in the system which could help to make system development very easy. Mainly, it will also help in improving the system and hence the error which may occur in the system could also be solved easily. It is done to make the main development phase easier and reliable.

## 2.2 Analysis Methodology

Whenever we try to analyze the analyze the system, we have to take the certain approach while developing the software. There are mainly three types of the approach, they are hard, soft and combined approach. I have chosen to take the hard system approach because it can also be reliable for the small scale as also well as for large complex system. This methodology will also follow a logical sequence of steps and have to follow the certain rules and guidelines which needs to be maintained. There are many types of Methodology within the hard system approach but SSADM is more appropriate for my system as it is very easy and reliable to use.

SSADM could be viewed as the three views where the method is break down into the three steps like process view, data view and event view. I have chosen the SSADM approach because it follows the waterfall model where all the steps must be completed before progression to the following step and since while developing the website, SSADM will help to manage the proper planning and scheduling. In this methodology We have to create DFD which will help to show the data flow among the various parts of the system, the components where the data are flowed are processes, data entry, data flow etc. Here is the data flow diagram of the given system and it will help to show all the processes and output withi the system when we are doing the hard system approach it is also the part of the SSADM.

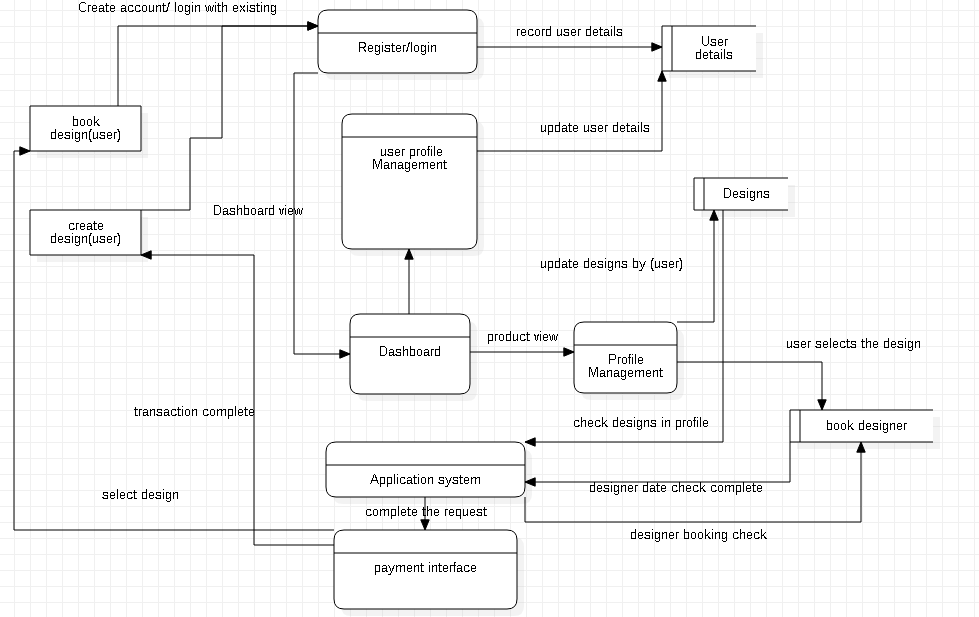


Figure data flow diagram

## 2.3 Feasibility Study

It is the part of analysis which will help to determine whether the project is legally, technically and financially achieved. This study will also help to know whether the project has positive or negative outcome before the starting of project. While doing feasibility study we have to check the each and every aspect of the feasibility like technical, financial etc. For having the better developed system, the system should fulfil every feasibility test. A good and proper feasibility study may reflect in better success of the project. There are few areas that are examined by feasibility study and regarded as the determining factor of the project (Mukund, 2017). Some of them are explained below:

• **Technical feasibility** : In this feasibility , We focus mainly on the availability of technical resource in the development phase of the following system. the technical resource such as, hardware, software and memory which will help in the development phase of the project. Since in my project, I have software as the technical resource. Hence, the project is technically feasible.

• **Social feasibility**: In this feasibility, It will help to determine the social factor such as political condition, environment and the scope or area that is being covered by project. Since in my project, there isn’t many social features but this project will affect the society directly or indirectly. In my project, the project coverage will cover only within the state.

**• Legal feasibility**: In this feasibility, It will help to investigate whether the following project is limited or within the boundary of legal requirements that needs to be fulfilled such as data protection law, copyright law etc. It will also help to develop the system by the legal constraints and rules. In my project, I haven’t use any functions and features that are illegal and I have also taken any consideration in any legal issues in the planning of the project and by this we could make the project more legally feasible.

**• Financial feasibility**: In this project, the following aspect is mainly dependent in the cost of the project that is needed while developing the project and it also considers the benefits or profits that is being earned by the development of the project. This feasibility will also help to determine the positive benefits in the project. Since I am developing my website without and cost or resource and chosen open source for the development, there aren’t any financial complication and hence is feasible.

## 2.4 SRS

SRS or software requirements specification can be defined as a notes or the description of the system that is going to be developed. For developing, there maybe functional and non-functional requirements which may include a various use cases which will describe the user interactions. The device and the system resource which is needed for the development of the project comes under the SRS. It also differs from the development methodology used and the type of software which should be developed. It also differs from the development methodology used and the type of software which should be developed.

## 2.4.1 Functional Requirements

The functional requirement can be defined as the declaration of the intended function which contains technical details , data manipulation which defines what the following system should achieved. The different functions which are included in the following system are shown in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FR ID** | **FR Description** | **Data required** | **Rational** | **Dependency** |
| FR1 | Admin signup | Name, username, password, email, phone number | Admin login data. |  |
| FR2 | Admin login | Username, password | Security and privacy. | FR1 |
| FR3 | Upload Designs | Name, price,  details | Uploading  Designs by admins | FR1, FR2 |
| FR4 | View Designs | Name, price, details | View the designs of  interior by admins | FR1, FR2 |
| FR5 | Update Designs | Name, price, details | Updating the designs of interior by admins | FR1, FR2, FR4 |
| FR6 | Delete Designs | Name, price, details | Deleting data of designs by admin | FR1, FR2, FR4 |
| FR7 | View user details | user,  username, email, | Vie the data of the user and modify it | FR1, FR2 |
| FR8 | Insert new designers | Designers name , Designers details | The list of designers | FR1, FR2 |
| FR9 | Manage users | Name of user,  username, email | Suspend and delete unwanted and untrusted users by admin | FR1, FR2, FR7 |
| FR10 | User registration | Name, username, password, email, phone number | Users login data. |  |
| FR11 | User login | username, password | Security and privacy. | FR10 |
| FR12 | View and book selected designs | Username, designs types, designs details | View the selected designs by the user | FR10, FR11 or FR1, FR2 |
| FR13 | Find filtered designs | Design type, designplace use designer | To find the selected designs from filteration | FR10, FR11, FR12 |
| FR14 | Book designers | Designers name , designers cost, details, date | Book the designers for interior design for the date | FR10, FR11, |
| FR15 | Manage profile | Name, username, password, email, phone number, | Update and change the personal data and details of user. | FR10, FR11 |
| FR16 | Chat with admins | Username, message. | Chat by users with admin for informative purpose. | FR10, FR11 |
| FR17 | Designer book schedule | Designer name, date | For designers booking for designs certain date | FR10, FR11, FR12 |
| FR18 | Payment for designers hire | Designers name , payment , payment type | For paying the designers | FR10, FR11, FR12, FR14 |
| FR19 | Payment management | Username, Designer name | For payment cash n meet or online | FR10, FR11, FR12,FR14 |

## 2.4.2 Non-functional Requirements

These are the requirement which help to describe whether the following system will operate and perform the requirements and user requirements. It will also help to provide the constraints to the behavior of the system. Non-functional requirements are very critical for the development of the following system as it will help to make system more robust, effective faster and user friendly. There are some of the non-functional requirements. Some of them are :-

|  |  |  |
| --- | --- | --- |
| **ID** | **NFR Title** | **NFR Description** |
| 1 | Security | It is the most important requirement in the development of the system which could be covered by using the strong authentication and encrypted algorithms |
| 2 | Performance | It is the factor which helps to show whether the system is fast or effective. We should use the updated technology to enhance the performance of the system. |
| 3 | User friendly | The system should be very easily usable and should be used easily by the user. By the help of the High user interface to would be more usable and user-friendly. |
| 4 | Reliability | The developed system should be highly reliable and hence should be easily accessible and less failures while operating the website. |
| 5 | Availability | The system should be easily available for the user that are using the website and hence the search engine should be optimized. |
| 6 | Scalable | The system should be more responsive and can be accessible from all the device. for example it should be accessed from mobile and laptop |
| 7 | Data Integrity | The data and content of the website must be secured. Only the authorized people can change the following data from the website. |
| 8 | Environmental condition | The external factor during the development of the system should be properly managed for example the pc should be up to date |
| 9 | Recoverability | This factor should be applied if the developed system have been accidently damaged or corrupted and through this aspect I could recover and get the backup of it |
| 10 | Capacity | The system should store the data as required and due to which the speed or the performance of the system doesn’t; degrade. |

## 2.4.3 Prioritization

Prioritization can be defined as the process of arrangement of the required fields and the features according to importance in the system. There are several requirements on the system and are arranged accordingly using the MoSCoW prioritization. Here are some of the prioritization according to the feature in the given below. I have also set the priority order and requirements.

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Features** | **Prioritization** |
| 1 | User/Admin registration | Must have |
| 2 | User/Admin Login system | Must have |
| 3 | Upload/update/delete Designs/Designers | Must have |
| 4 | Manage users | should have |
| 5 | Inform users about Designs | Should have |
| 6 | Chat with admins | Would have |
| 7 | Manage profile | Could have |
| 8 | Book Designs for interior | Must have |
| 9 | Book Designers for Designing | Must have |
| 9 | Find Designs | Could have |
| 10 | View user details | Would have |
| 11 | Designers payment | Must have |
| 12 | Designers details | Would have |
| 13 | Find Designers | Could have |

## 2.4.4 Hardware/Software Specification

There were some hardware and software specification while designing and development of a project. The hardware and software specification for my project is provided below:

**Hardware Specifications:**

* Processor: Celeron 1000MHz or any Pentium processor
* RAM: 2GB or higher
* Hard disk: 30 GB or higher
* Display type: Standard VGA or SVGA card
* Peripherals: keyboard, mouse.

**Software Specification:**

* Operating system: windows 7 or higher
* Front-end: bootstrap
* Back-end: PHP, MySQL, XAMPP

## 2.5 Use Case Diagram

Use case can be defined as the diagrammatical representation of the various interaction details within the system. It will help to show the relation among various actors, system and the use cases. Use case would contain all the activities within the system that is important to the user. In the following user case, the actors are connected with other actor by the help of use case within the system. The main components of use case are:-

* Actors: It is the people who is employed and it is represented by the stick figures
* Use cases: It helps to know the roles of each actor in the following system
* Association: It will help to refer the relation of actors with the related use cases
* System Boundary: It is refered to rectangular box where the system is posted

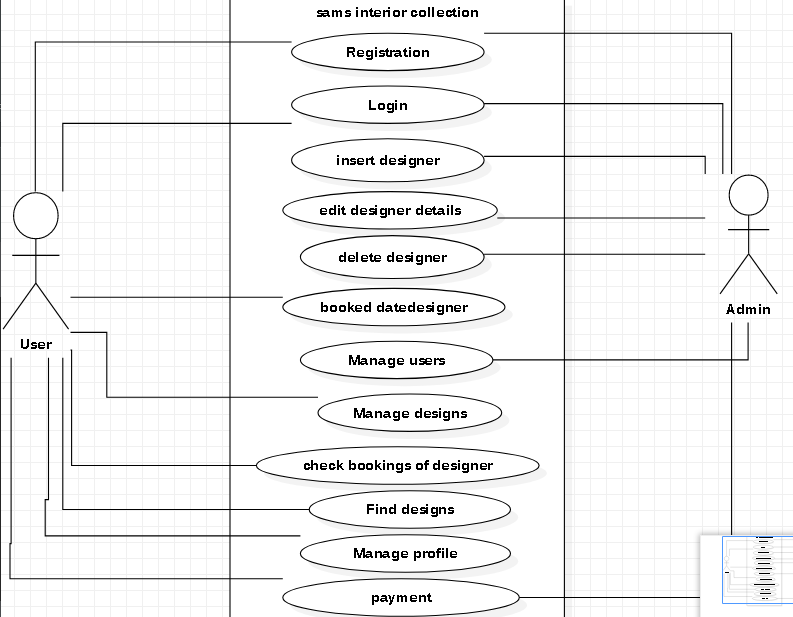


Figure use case diagram 1

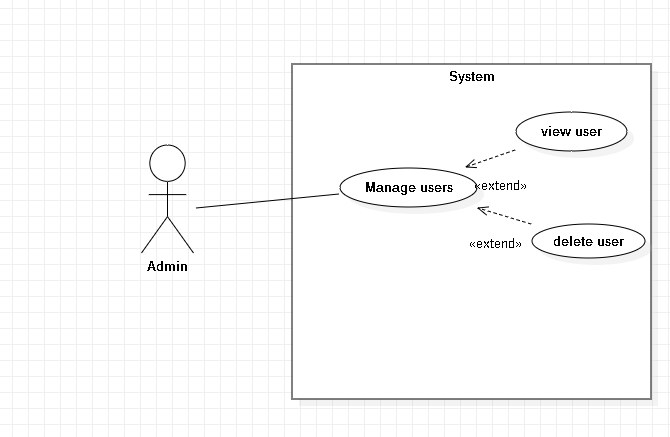


Figure fig3 use case

|  |  |
| --- | --- |
| Justification | This use case shows that the admin can manage the user by viewing and deleteiing the user. |
| Actors | Admin |
| Supporting actors | N/A |
| Primary flow | 1. Admin logs into the system 2. Admin views the user 3. Admin will delete the user if something goes wrong |

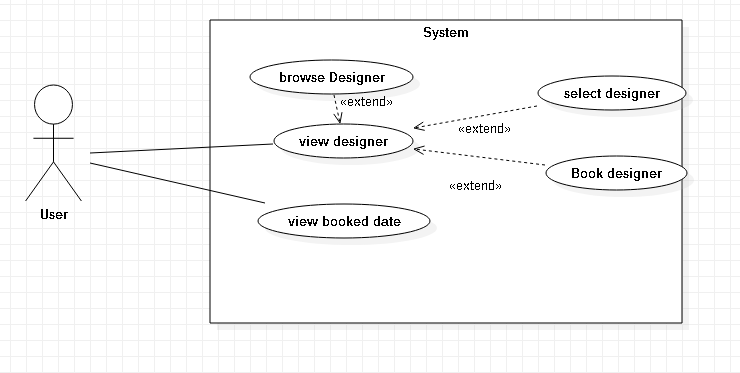


Figure use case

|  |  |
| --- | --- |
| Justification | The user can choose the designer for the designing and can view the details of the designer and can view the booked date of the designer |
| Actors | User |
| Supporting actors | N/A |
| Primary flow | 1. User search the designer to implement the design 2. User will select and book the designer if he is free by checking the booked |
| Secondary flow | 1. After viewing the designer we can select and book the designer for implementation |

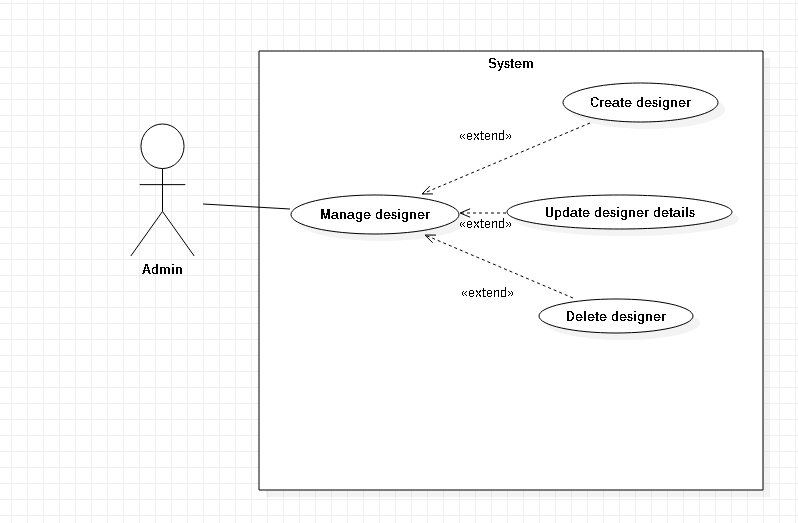


Figure use case

|  |  |
| --- | --- |
| Justification | This use case shows that the admin can manage the user by creating and adding the designer as well as update the designer details and can also delete the designer |
| Actors | Admin |
| Supporting actors | N/A |
| Primary flow | 1. Admin logs into the system 2. Admin can delete and add the user 3. Admin could update details of the designer |

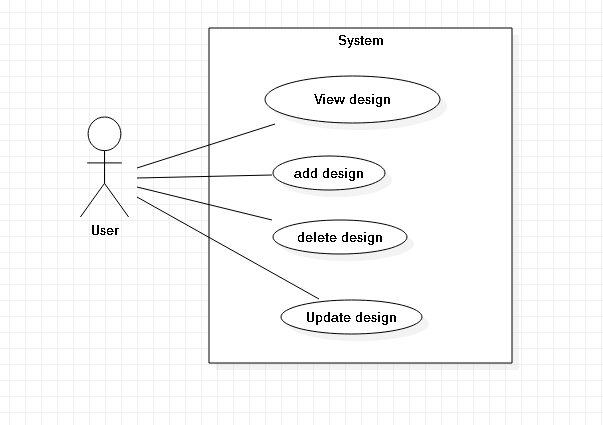
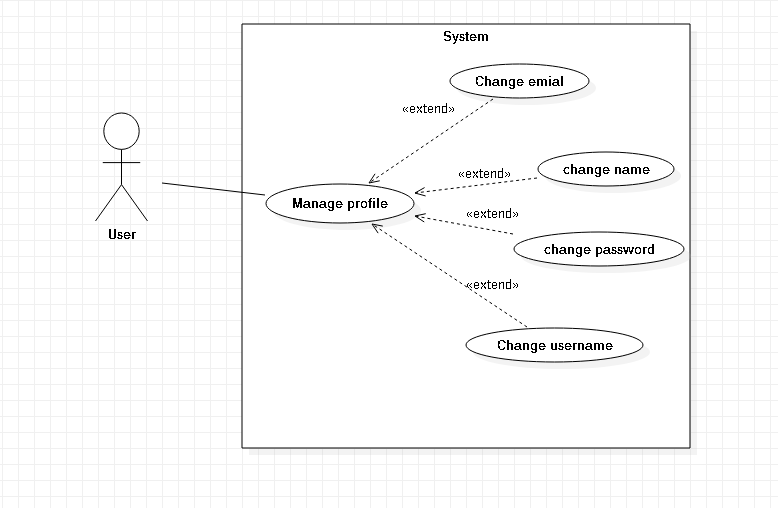


Figure ise case

|  |  |
| --- | --- |
| Justification | This use case shows that user can view as well as add delete and update the design |
| Actors | User |
| Supporting actors | N/A |
| Primary flow | 1. User can view their designas well as their own design 2. Userwill add the new design 3. User will update the existing design 4. User will delete the existing design |



|  |  |
| --- | --- |
| Justification | This use case shows that user could update their profile by changing and updating their details |
| Actors | User |
| Supporting actors | N/A |
| Primary flow | 1. User will be logged in 2. User will change the name, email and password after getting logged in. |

The following above diagrams shows the use case of the following system. We could see the interaction between two actors like user and admins in the use case diagram. The main use case diagram is showing the interaction between user and admins in the following system. In the other use case, it is showing the behavior of the actors and the interactions.

## 2.6 NLA

Natural language analysis or NLA is the process of breaking the phrase in the form of noun verb and adjectives. But while we are developing the system we break the system into classes, function and attributes where classes acts as a nouns, verbs as a functions and attributes as a adjective. Here are some of the example of NLA about the following system.

|  |  |  |
| --- | --- | --- |
| Nouns(classes) | Verb (Function) | Adjective (Attributes) |
| Admin | Signup, login, view designs, view designers, add designers | Admin\_id, admin\_name |
| User | Book designer, view designers, select designs | User\_id, username, email |
| designs | Upload\_design  Edit\_design  Delete design  View design | design\_id, design\_type, design\_roomtype |
| designer | View designer  View book schedule  Book designer | Date time, username |
| Order\_designs | Book design, select designer delete design | Username, designs, ordered date |

## 2.6 Class diagram

Class diagram can be known as the structural diagram of the system including various classes, attributes and other opeations. It will help us to provide the desig of the system to show the relationship between various classes within the system. Hence to clarify, I have created the class diagram by the help of the Natural Language analysis. The following class diagrams are:-

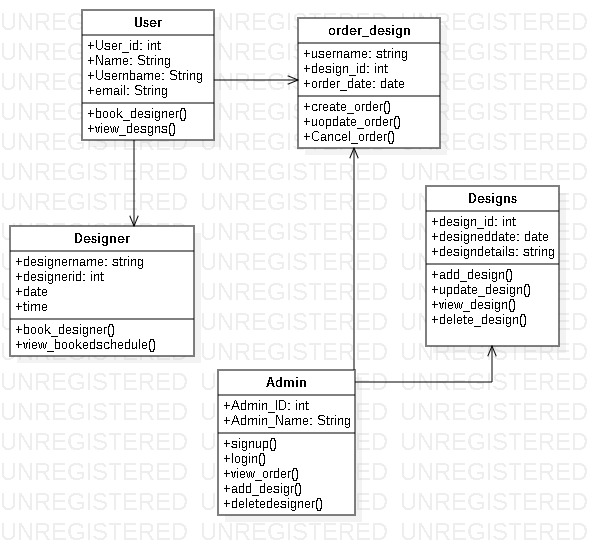


Figure Class diagram

In the following class diagram, I have created the classes known as designers, admin, designs, ordered designs and user. I have also concluded various function and their attributes which will help to identify their function very easily. By the help of this class diagram, I could do further task more easily.