# Chapter 2: Analysis

It consists of different subtopic which are mention below:

1. Introduction
2. Analysis methodology
3. Feasibility study
4. Information gathering
5. Software requirement Specification
6. Use case
7. System Architecture

## 2.1 Introduction:

Analysis is the process of gathering and specifying the requirement on the basis of the priority of that function, identifying the problems and split into parts/ components so that it can be easily solved.

It is conducted for the purpose of studying the products/ systems or its parts in order to find out the objectives. In other hand it helps to improve the system and make sure that all the parts of the system work effectively to accomplish their purpose. It specifies what are function should be in the system and what the system should perform.

Survey

Questioner

Economical

Object oriented analysis

System architecture

Use-case diagram

SRS

MoSCoW

Non functional

Functional

Time

Legal

Technical

Ethical

Analysis methodology

Feasibility study

Information gathering

Requirement specification

Use case and system architecture

## 2.2 Analysis methodology (Object oriented Methodology)

For the analysis methodology I have used object-oriented analysis. Object-oriented analysis is technical approach of analysis of system or application by the use of object-oriented paradigm and visual modeling. It does not only focus on the process or data of the system but view the system or application as a collection of interacting objects which works together to perform certain task.

Object-oriented analysis uses three analysis techniques which are given below:

1. Object modeling
2. Functional modeling
3. Dynamic modeling

### 2.2.1 Object modeling

It identifies objects, classes, their relationships and methods and attributes. It represents the static structure of the product/ system. Class diagram is the representation of object model.

### 2.2.2 Dynamic modeling

Dynamic modeling identifies the behavior of the system over time and flow of control and events. Event trace diagram and state transition Diagram represent the dynamic model.

### 2.2.3 Functional modeling

Functional models are the sets of DFDs (Data flow diagrams) that identifies the internal processes are independently from how these processes are performed.

## 2.3 Feasibility study

Feasibility study is analysis process in which all the factors like economic, technical, legal, ethical and scheduling of the project is studied to find out whether the project is feasible to procced. If the strength of project overcomes the weakness of the project and if opportunities overcome the threat of the system then the project is feasible to do.

1. **Economic feasibility:**

Economic feasibility deals with the cost/ benefits analysis of the system/ project. It will check whether the project budget is sufficient for the cost of development of this project. If it is sufficient enough then the project is proceeded. If not then the project will compromise some function/ features or even drop. While seeing the **Doctor information system,** total cost of the development is sufficient enough for the budget.

1. **Technical feasibility:**

Technical feasibility study focuses on the technical aspects like technical resources used, tools used for the development. I have used XAMPP, Nodejs, git bash, sublime text editor etc. while developing the project which are hardware compatible and did not need more space/ memory for installation. All this tool used are free of cost and hardware compatible so, I think this project is technically feasible.

1. **Ethical feasibility:**

Ethical feasibility deals with all the social, cultural, environmental, ethical aspects of the project. This is not developed for the specific religion so anyone can use it. Even I am not going to use specific cultural picture/ images or any other stuff. And it cannot create any social conflict. So, I think it will be socially, culturally, and ethically feasible to do.

1. **Legal feasibility:**

Every tool used are legally available for free. So, it will not cause any legal issue in coming future. Doctor information system which I’m going to create will follow the law of country for the development of product. Therefore, it will not cause problem while developing the product. So, I think the system will be legally feasible.

1. **Schedule feasibility:**

It deals with the timing feasibility of the project like does the project end in time? Does it include all the features/ functionality in time period provided for each functionality? This project will not have bulky and complex functionality. It has 2/3 days extra for the parts of development so, I think there will not problem in time management. So, this project is time wise feasible to do.

## 2.4 Information gathering:

It is the process of gathering requirement and information of the user interface with the help of different methodology like interview, survey, questionnaires, observations, study of the existing documents, etc. Information gather will properly documented, well defined and should be taken to the appropriate group. So, that it can possess some quality information for the well development of the system. For the information gathering I have used Documentation and questionnaire.

### 2.4.1 Questionnaire/ survey

It is the process of gathering information to the large number of people with the help of some sorts of sample questions. Question sample is given below:

**Questions:**

1. Name:
2. Age: [ ] below 20 [ ] between 20-40 [ ]between 41 -60 [ ] 60 above
3. Religion:
4. Education level:
5. Address:
6. Occupations:
7. How often you use internet?

[ ] once a week [ ] daily [ ] once a month [ ] on the basis of need

1. How often you visit health related websites?

[ ] once a week [ ] daily [ ] once a month [ ] on the basis of need

1. How often you provide some queries or comment on health-related post?

[ ] sometime [ ] never [ ] on the basis of need

1. Did you like see diseases related newsfeed/ post?

[ ] yes [ ] no

1. How often you visit to doctors for the regular checkup?

[ ] Regular [ ] when fell sick

1. What types of interface did you like to have in such web applications?
2. Did you provide feedback to the doctor’s services after the consultant/ checkup?

[ ] Yes [ ] No

1. Did you like to online consult with doctors?

[ ] Yes [ ] No

1. Which type of color should be in the background of such web base application?

…………………………………………………………………………………………..

1. Any feedback that you like to provide

…………………………………………………………………………………………….

Thank you!

**Reason for using questionnaire:**

🡪 the system going to develop is for large number of people so, it will not be appropriate to use interview with people.

🡪 save time and efforts

🡪 reliable

### 2.4.2 Documentation

I have visited a lot of website for the research of functionality that should be included in Doctor information system.

## 2.5 Software requirement specification

In the software requirement specification, I have used some method to prioritize and identifies the requirement gather for the doctor information system. The method that are used are given below:

1. **Functional requirement:**

Functional requirements are those requirements which must be included in the web application. We must call that requirements are the foundation of the system. These describes systems behaviors. Some of them are given in the table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| F-ID | Title | Dependency | Rational | Description |
| F001 | Registration | No | Security | Provide email, password and other information. |
| F002 | Login | Yes [registration] | Security | Provides email and password |
| F003 | Show profile | Yes [F001] | Availability | Show provide information during registration |
| F004 | Search doctors by choosing locations | Yes [F001] | Availability | It is dependent on the information provided by the doctors while registration |
| F005 | Update doctors’ profile | Yes [F001] | Increase reliability and accuracy | It is dependent upon doctor id for the changes. |
| F006 | Delete information by admin | No | Increase reliability | If some doctor is fraud then their profile should be deleted |
| F007 | Doctor can add post | No | Increase interaction to system | Doctor can add some health-related post so that improve the knowledge of others users. |
| F008 | User can update their profile | Yes [F001] | reliability | It increases interaction of the users to the system |
| F009 | User can provide feedback | No | Involvement | They can provide feed on what they like? what they hate of the system? what should be improve? |

1. **Nonfunctional requirements:**

Those requirement without which application should be just working but not well. These all the function make more decorative and make our product sustain in the market. Some of them are mention below in the table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NF-ID | Title | Dependency | Rational | Description |
| NF001 | Making system secure | Non | Provide security | It helps to increase the level trust |
| NF002 | Available when require | Non | Increase reliability | It makes system reliable |
| NF003 | Scalable | Non | Increase the productivity |  |
| NF004 | Maintainable | Non | Sustainability of the system |  |
| NF005 | Usability | Non | Make more usable to the users |  |
| NF006 | Efficiency | Non | Use less resource and search faster |  |

1. **MoSCoW prioritize:**

MoSCoW function is invented by “Dai Clegg” from the software company Oracle. MoSCoW prioritize is about setting the requirements by the order of priority. To deliver right working product at right time so, that client can use the working software. It makes easier and clearer. What function should be done first?

Every capital letter of MoSCoW has its own meaning. Full form of the MoSCoW is given below:

* Must have: those requirements which should give more priority during development. These all are implemented first and help to make working and without its software will be useless.
* Should have: these are less important than “Must have” function but these should be later after completing must have functions. These are like supports of the foundation of the product.
* Could have: It should be done if we have enough time if not then it will be implemented in the next version.
* Won’t have: these features should not be included in the system.

MoSCoW prioritize table is mention below:

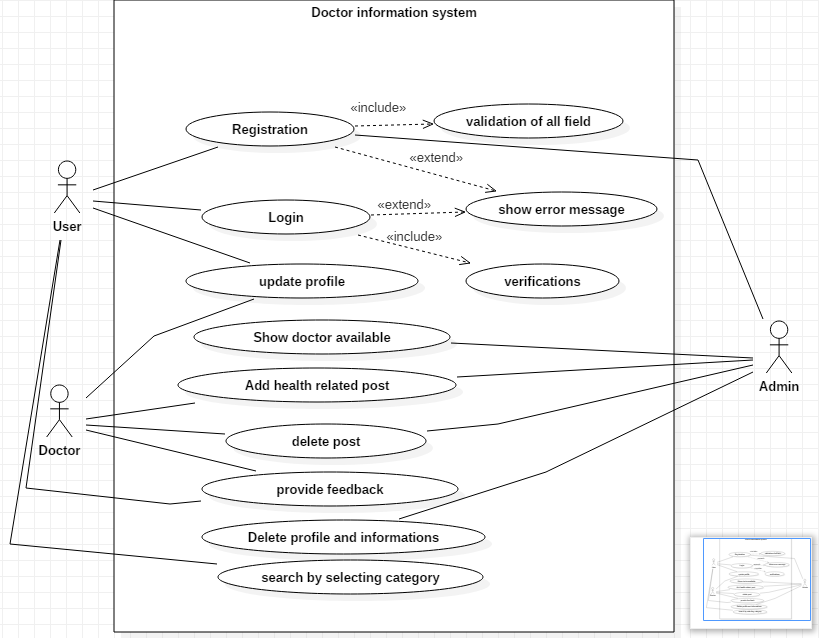
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| M-ID | Title | MoSCoW | Rational | Remarks |
| M001 | Register | Must have | Security |  |
| M002 | Login | Must have | Security |  |
| M003 | Show error message | Should have | Informative |  |
| M004 | Login attempt | Should have | Security |  |
| M005 | Show doctors list | Must have | Availability |  |
| M006 | Admin can another admin and provide authority | Could have | Increase efficiency |  |
| M007 | User’s provide feedback to doctors | Could have | User interaction |  |
| M008 | Doctor can post health related news | Should have | Informative |  |
| M009 | User provide rating to the doctors that they have consulted | Could have | Increase interaction |  |
| M010 | Doctor can reply the message post by users | Could have | User satisfaction |  |
| M011 | User can buy premium consulting membership | Could have | Increase efficiency |  |
| M012 | Reliable | Should have | Increase satisfactions |  |
| M013 | Security | Should have | Increase privacy |  |
| M014 | Available | Should have | Increase user participations |  |
| M015 | Scalable | Should have | Increase productivity and efficiency |  |
| M016 | User update their profile | Should have | Increase user interaction |  |
| M017 | Doctor update profile information | Should have | Increase doctor interaction |  |
| M018 | Admin delete the doctor information if not available | Should have | Reliability |  |
| M019 | User comment to the doctor post | Could have | Feedback |  |
| M020 | Search by hospital | Could have | Select category |  |
| M021 | Search by selecting doctor specialist | Could have | Increase efficiency |  |
| M022 | Show the users past consulting history | Could have | Reliability |  |
| M023 | Online consulting | Could have | Interaction |  |

## 2.6 Use case diagram

Use case diagram show the user interaction to the system. There are some internal and external factor for making the interaction to the system. These external agents are known as the actors.

Purpose of using Use Case Diagram are mention below:

* It is used to gather the requirement of Doctor information system.
* It is used to show outside view of the system.
* It shows the interaction of actors to the functionality
* It helps to determine actor which is going to influence the system

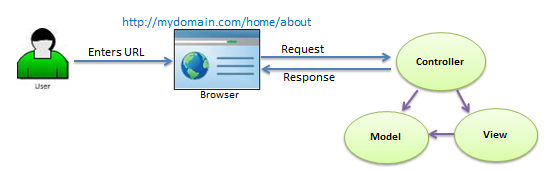


## 2.6 System Architecture

### 2.6.1 Design pattern

I will like to use MVC (Model View Controller) design pattern for this project. (Chakraborty, 2018) In MVC design pattern whole project will be divided in three main logical component Model, View and Controller. Model component is related to all the data-related to logic that user works with. View component is related to all the UI logic. Controller works intermediate between model and view. It handles the user request by processing data logic (model) and interact with view.





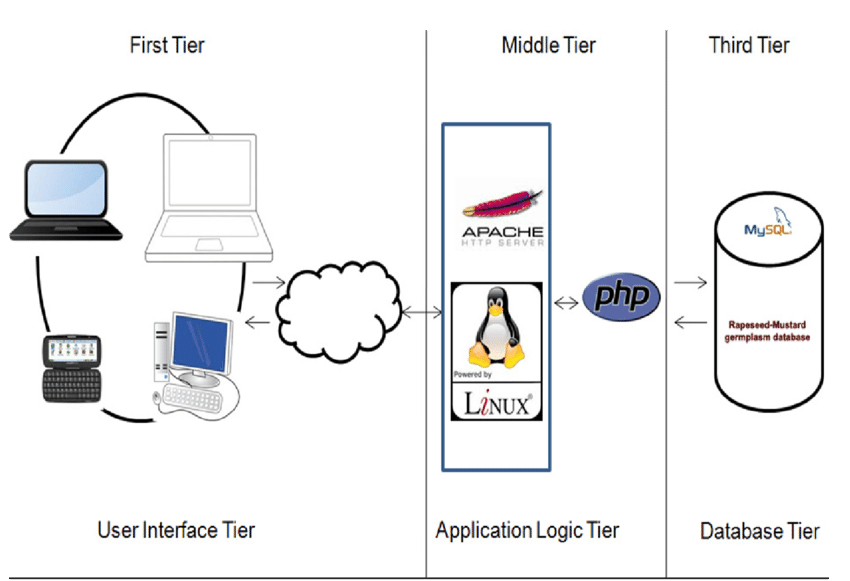
I will be using this design pattern because of these reasons which are given below;

* It decreases the complexity of program.
* It increases the maintainability of the program.
* We can easily find out where is the problem while something went wrong.

### 2.6.2 System architecture definition

System architecture is the conceptual model that determines behavior, structure of the system. For the system architecture I will like to use **3-tier** architecture because a tier can be changed or relocate without affecting other tier and makes easier to continually evolve the application as new need or requirement and opportunities arises. 3-tier architecture is modular client server architecture that consists of a presentation tier, data tire, application tier. (Gilbert & Rouse, 2019)

* **Presentation tier:** this tier is built with HTML5, CSS and JavaScript are deployed to computing device with web browser. It communicates with other tiers with the help of API (Application Program Interface) call.
* **Data tier:** Data tier is referred as storage tier. It consists of database and program for performing data read and write operation.
* **Application tier:** it is also known as logical tier which is written in programming language such C#, java and contains business logic that can hold up application’s core function.



### 2.6.3 Scenario

“The Brothers” Company is a software company established in 2010, located at Westgate Court, Westgate St, Cardiff CF10 1DD, UK. This company is hire you to provide working web application which should have some features.

For users:

* User can be able to login to the system
* User can be able to register to the system
* User can be able to provide feedback
* User can be able to search the doctor located in his community by selecting the location
* User can be able to update his information

For doctors:

* Can be able to register to the system
* Able to add the post
* Able to delete the post
* Able to update the post
* Able login to system
* Login attempt will not more than 3 times
* Update profile information

System:

* Admin login
* Login attempt will not be more than 3 times
* View the list of doctors
* View the post
* Add the post
* Update the post
* Delete the post
* Delete the doctor information/ account if not valid
* Show the users details

#### 2.6.3.1 NLA (Natural Language Analysis)

It is the process of identifying verbs, nouns and adjectives from the given scenario or descriptive text.

* Nouns are related with possible candidate class
* Verbs are related to possible candidate methods
* Adjectives are related to attributes

**Candidate method selection:**

Candidate methods are those methods which should be include to the system for the development of the system.

**Verbs:**

Add, delete, login, register, provide, select, located, established, hire, attempt, show, View

Candidate method selection are listed below:

|  |  |  |  |
| --- | --- | --- | --- |
| V-ID | Verb | Candidate method | Description |
| V001 | Add | Yes | It helps to create the post which will be more useful to the system to implement |
| V002 | Delete | Yes | It will used to delete not useful information and post |
| V003 | Update | Yes | It is used to update information by the all the users, doctors, admin |
| V004 | Login | Yes | It is one of the most useful function of any system to add the security level |
| V005 | Register | Yes | It is another useful function |
| V006 | Provide | No | It is not specific |
| V007 | Select | Yes | It helps to select the category |
| V008 | Located | No | It is out of scope or need not to mention |
| V009 | Established | No | It is out of scope or not useable function throughout the project |
| V010 | Hire | No | Out of scope |
| V011 | Show | No | Similar to view |
| V012 | Attempt | Yes | It will increase the security level |
| V013 | View | Yes | It will show the information like if user want to view his/her profile, doctor list, post |

Candidate method are given below:

|  |  |
| --- | --- |
| S. N | Candidate method |
| 1 | Add |
| 2 | Delete |
| 3 | Update |
| 4 | Login |
| 5 | Register |
| 6 | Select |
| 7 | Attempt |
| 8 | View |

**Candidate class selection:**

Those classes which are fundamental of this system are known as candidate classes.

**Nouns lists:**

Company, software, web application, users, system, user, post, doctor, community, feedback, information, admin

Candidate class selection are given below in the table:

|  |  |  |  |
| --- | --- | --- | --- |
| C-ID | Noun | Candidate class | Descriptions |
| C001 | Company | No | It is out of scope |
| C002 | Software | No | It is not related to the system and it is out of scope. |
| C003 | Web application | No | It is not required for the class |
| C004 | Users | No | It is better to use singular form |
| C005 | System | No | It does not provide specific meaning |
| C006 | User | Yes | It will be appropriate to user as customers of web-based application |
| C007 | Post | Yes | It will contain information of who posted and health related information. |
| C008 | Doctor | Yes | It is type of user but have different role it is better to use it as different class. |
| C009 | Community | No | It does not have specific meaning |
| C010 | Feedback | Yes | For the better normalization it useful to make feedback as class. |
| C011 | Information | No | No specific meaning and cannot be able use as class |
| C012 | Admin | Yes | It should be added as class because its main functionality to handle the system. |

Candidate classes are given below:

|  |  |
| --- | --- |
| S. N | Candidate class |
| 1 | User |
| 2 | Post |
| 3 | Doctor |
| 4 | Feedback |
| 5 | Admin |

Initial class diagram is given below:

