Chapter 1

Introduction

* 1. **Background of system**

As we are living in the era of technology, this day’s people are very modern and technology based. We people do not like stand on long queue and wait for the turn. Everyone use technology like laptop, mobile phone, and internet for taking appointment or heath suggestions with doctor online. The proposed project “city health care website” whereby patient and doctor can share data in a safe way.

“City health care website” through the help of this website patient can share their health issue with the doctors in a safe way. Patient can also take health tips from the doctors. They can also know at what time doctors are available at the city health care and as per the schedule, they can take their appointment with the individual doctors.

You just need to register in the web and login to make appointment with the individual doctor. Patient can take appointment or share their health issue with the doctors 24hours a day from any part of the globe.

**2.2 Over view of the system**

To the small website a small project city health care website the framework is worked in the PHP programming language and its supporting apparatuses. The internet is fabricated utilizing HTML and CSS. Advance GUI highlights and some approvals is finished utilizing java script. The fundamental rational is composed utilizing PHP language. The information is put away in MYSQL database. The frame work is facilitated locally in apache server utilizing XAMP control panel. The coding is done in the content tool subline text. This system is developed for city heath care website so that patient can get some free health tips and take appointment of the doctors online from home or from any part of the country by via internet.

* 1. **About the system**

City health care website is developed to manage the system of city health care and for the easiness of the patient. Through the website they can take free health tips and take appointment of the respective doctor online. User need to register for creating their account in order to take appointment online. After registration user can direct login into website.

* 1. **Aims and objectives**

The task scope characterizes the greater part of the work to be finished amid the venture. The degree portrays what is to be expert with the undertaking. It characterizes the finished result to be conveyed to the client by a specific point in time and at a predetermined spending plan or cost. Clients can take free health tips and can take appointment of the respective doctors. It's quick and dependable approach to consult with the respective doctors without wasting time. The client will have the capacity to utilize the application. The given points are the aims of my project:

* Aim of this project is to demonstrate that with better interactive features.
* It is easier and quicker.
* To manage user data in efficient and effective way.

The objective are as follows:

* User can take the appointment of the doctor from anywhere.
* User can see which doctor is available at what time.
* Patient can share their health issue with doctor online.
* To secure information in database.
* Data redundancy should be reduced.
* To develop the system based on the designs produced which will help to fulfil to meet the requirements.

Chapter 2 Analysis

2.1 Introduction

The initial stage in structure a framework is to comprehend what that framework should be. Investigation is the initial step which is bridge to accumulate the data. Through the investigation procedure it is straightforward the what framework ought to do. The explanation behind investigation is to find the necessities which recognizes and arrange the information need that causes the framework to get the framework objective. And furthermore, portrayal of all parts of the present data framework.

**Rich picture**

A Rich Picture is a way to explore, acknowledge and define a situation and express it through diagrams to create a preliminary mental model. A rich picture helps to open discussion and come to a broad, shared understanding of a situation.

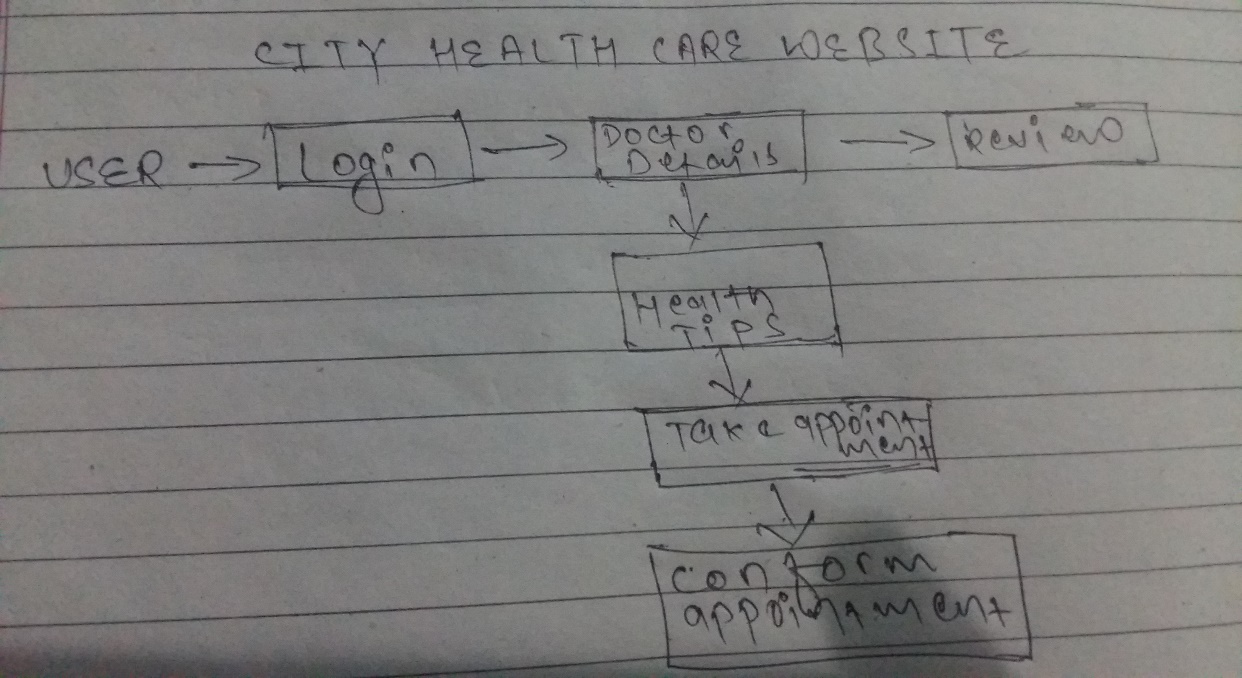


Fig1: Rich picture.

2.2 Feasibility study

The word ‘**feasibility** ‘means the degree or state of being easily, conveniently, or reasonably done. If something is ‘**feasible**,’ it means that we can do it, make it, or achieve it. In other words, it is doable and also viable.

A **feasibility study** is an evaluation and analysis of a project or system that somebody has proposed. We also call it a **feasibility analysis**. The study tries to determine whether the project is technically and financially feasible, i.e., is it technically or financially viable? Financially feasible, in this context, means whether the project is feasible within the estimated cost.

Types of feasibility study

1. **Economic feasibility**

Economic feasibility analysis is the most commonly used method for determining the efficiency of a new project. It is also known as cost analysis. It helps in identifying profit against investment expected from a project. Cost and time are the most essential factors involved in this field of study.

1. **Technical feasibility**

Technical feasibility study is the complete study of the project in terms of input, processes, output, fields, programs and procedures. It is a very effective tool for long term planning and trouble shooting. The technical feasibility study should most essentially support the financial information of an organization.

1. **Operational feasibility**

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project.

1. **Schedule feasibility**

The process of assessing the degree to which the potential time frame and completion dates for all major activities within a project meet organizational deadlines and constraints for affecting change.

How long a project will run, analyzed by the schedule feasibility study?

1. **Legal feasibility.**

It determines whether the proposed system conflicts with legal requirements, e.g., a data processing system must comply with the local data protection regulations and if the proposed venture is acceptable in accordance to the laws of the land.

2.3 Requirement analysis

Functional requirements

A functional requirement explains what a framework ought to do. It portrays the primary practices of the item that the client anticipates from the framework. Capacities a framework should achieve like counts, information control, information preparing and so on are useful necessities.

Functional Requirements of my product are given below: -

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fid** | **Functions** | **Data** | **Rational** | **Dependency** | **Remarks** |
| F01 | Registration | Personal details | User information | F01 | Creates user account |
| F02 | Login | Email and password | Security | F01 | Open dashboard |
| F03 | Reset Password | Email | Update password | F02 | New password to access the profile |
| F04 | Post Notice | - | To post notice for user and organization | F02 | Notice will be shown to organization members and users |
| F05 | Healthy health tips | - | Health tips for user from doctors | F02 | User can follow health tips given by doctors |
| F06 | Take doctor’s appointment | Patient details | To consult with respective doctors | F02 | Patient can take appointment with doctors |
| F07 | Asking Questions regarding health problems | health topic Details | patient to respective doctors | F02 | user can ask about health problem to respective doctors |
| F08 | Answer the Questions  Regarding health | Health Question Details | doctors to user | F02 | Respective doctors can answer to user |
| F09 | Chatting | - | patient | F02 | patient can chat |
| F010 | Admin Login | Admin Details | Admin | F09 | Open Admin Dashboard |
| F011 | Admin Profile Update | Admin Details | Admin | F09 | Admin can update profile |
| F012 | Logout | After finishing the task user and admin can logout | End of the session | F02 | User and admin can logout |

Non-functional requirement

Nonfunctional requirements are vital to the success of software systems. If they are not properly addressed, undesirable results occur such as unsatisfied users, developers, and clients, and schedule and budget overruns to correct the software that was developed without the nonfunctional requirements in mind. Non-functional requirements generally specify the system’s quality attributes or characteristics.

Non-functional Requirements of my product are given below: -

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NID** | **Functions** | **Data** | **Rational** | **Dependency** | **Remarks** |
| N01 | Responsive | N/A | Support Different Resolutions | N01 | fits in different screen sizes automatically. |
| N02 | Usability | N/A | Easy to use and user friendly | N02, N01 | User friendly. |
| N03 | Maintainability | N/A | Easy to change | N03 | Easy Maintenance and optimization |
| N04 | Reliable | N/A | Accurate output | N01, N02 | Gives possible output as per input given |
| N05 | Robust | N/A | Supports many platforms | N05 | can work in different platforms |
| N06 | Multi-Browser support | N/A | Tested in more than two browsers | N02, N01 | Able to run in different browsers. |
| N07 | Scalability | N/A | To be able to handle workloads | N02 | Easy to handle the data flow. |

Moscow (must, should, could, won’t) prioritization

The Moscow strategy is a prioritization method utilized in the executives, business investigation, venture the board, and programming advancement to achieve a typical comprehension with partners on the significance they place on the conveyance of every prerequisite; it is otherwise called MoSCoW prioritization or MoSCoW examination.

The categories are typically understood as:

**Must have:** This point depicts in the last game plan necessities that must be fulfilled. Errand will be disillusionment if the necessities are not executed. It is the most essential things where the requirements are non-easy to refute.

**Should have:** An exceptional need highlight that isn't fundamental to dispatch. Here, the things we consider as critical anyway not imperative.

**Could have:** It is an alluring essential however a bit much. in the event that the undertaking's timescales are at peril this point will be eradicated from first degree.

**Won’t have:** In Current release this basic won't executed anyway may in future it will be consolidated into time of headway. these sorts of necessities don't hamper to be adventure successful.

For Functional Requirements

|  |  |  |
| --- | --- | --- |
| **ID** | **Functionalities** | **Priority** |
| F01 | Registration | Must have |
| F02 | Login | Must have |
| F03 | Reset Password | Must have |
| F04 | Post Notice | Must have |
| F05 | Healthy health tips | Must have |
| F06 | Take appointment | Should have |
| F07 | Asking Questions regarding health problems | Must have |
| F08 | Answer the Questions  Regarding health | Must have |
| F09 | Chatting | Should have |
| F010 | Admin Login | Must have |
| F011 | Admin Profile Update | Should have |
| F012 | Logout | Should have |

For non-functional requirements

|  |  |  |
| --- | --- | --- |
| **ID** | **Functionalities** | **Priority** |
| N01 | Responsive | Must have |
| N02 | Usability | Must have |
| N03 | Maintainability | Must have |
| N04 | Reliable | Should have |
| N05 | Robust | Must have |
| N06 | Multi-Browser support | Must have |
| N07 | Scalability | Could have |

Use case diagram

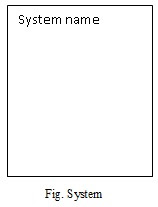
The use case diagram is a tool which plays a huge job in managing our reflection. It empowers us to speak to the wide association between the framework parts. Usefulness of the framework ought to be caught through the utilization case. They are utilized to gather the framework prerequisites containing inside and outside affected.

Given points are the purpose of use case diagram as:

* Demonstrates how actor deals with the framework.
* Gets the dynamic piece of the framework.
* The content of the system is determining.
* Framework architecture is validated.

These points are the fundamental images which are used in use case as: System: - It helps in the joint effort between one or more external entities in the system.

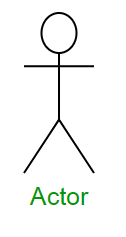
**System**: - By utilizing rectangle shape draw our limits of the system which contains the use cases. Actor ought to be put outside the framework boundaries.



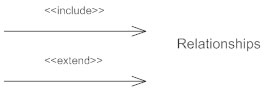
* **Use case**: - Use case is oval in structure which indicates various uses that an actor might have. It is represented by the symbol.



* **Actors**: - Actors are something that deals with our system. It is represented by the symbol.



* **Relationship**: Relationship can be named by arrow either uses or extends among use cases. To play out an undertaking on use case is required by another is shows by uses relationship where extends demonstrates optional under a specific use case.



Now, I have produced a Use-case diagram for city health care website.

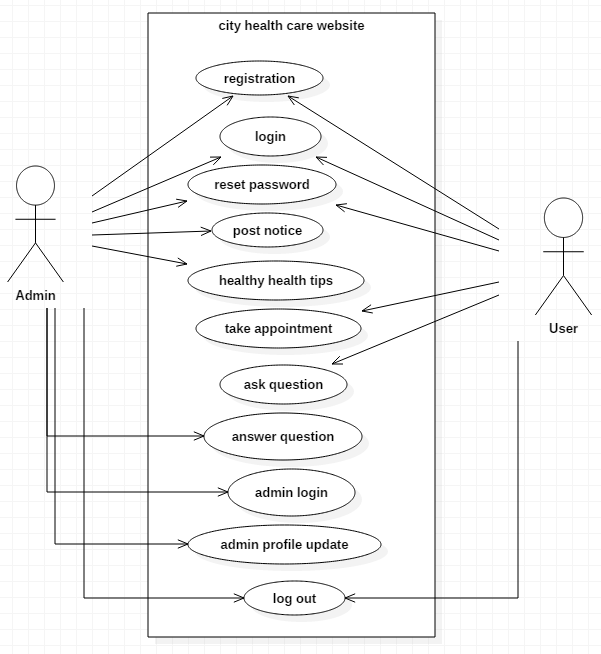


Fig: use-case diagram.

Natural language analysis (NLA)

Natural Language Analysis is a helpful tool. But it's not required. It works by performing in all respects basically linguistical extractions on a problem statement. Finding out noun, verbs and adjective. This fills in as the reason for your first draft of the class diagram. It’s an iterative process. You use it to comprehend the system.

|  |  |
| --- | --- |
| **Classes** | **Functionality** |
| login, registration, user, doctors, appointment, admin, | user registration, user login, select doctor, take appointment, update profile, admin login, post notice, view order, view user, view appointments. |

Initial class diagram

Framework static view is representing by class diagram. It indicates the classes with in a model. By utilizing the results of the crc procedure initial class diagram is created. It can represent to every one of these things effectively. Symbol of the class is basically rectangle shapes which is isolated into the three parts.

* The upper compartments show the name of the class.
* The center compartment shows list of attributes.
* The base compartment shows list of operations (Methods).

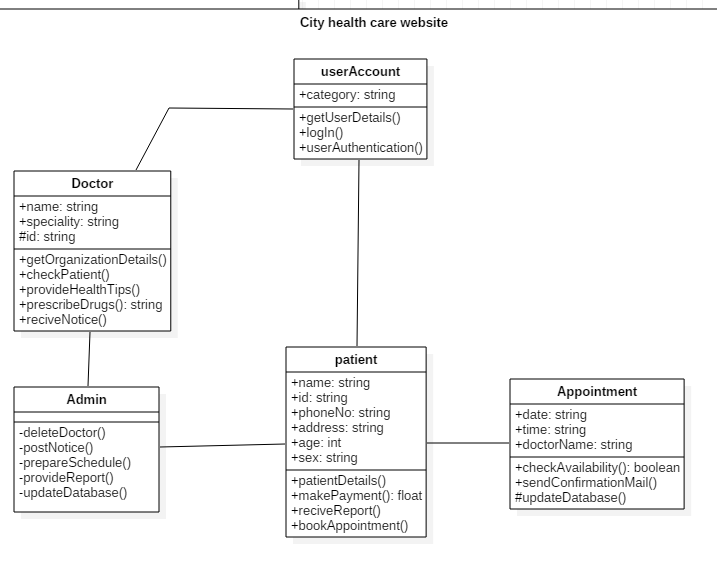
Relationships that are shown in the class diagram are as follows:

**Association:** It shows the instance variable which contains another object reference. Affiliation are shown as thin line connecting two class. Assortment appears by the affiliation.

**Aggregation:** It is an arrangement of affiliation relationship. It Is indicated by symbol hollow diamond pointing towards the entire.

**Composition:** It is again an entire or section relationship where on the off chance that the entire is crushed, at that point the area can't exist freely. A class can simply have a place with one arrangement at some random minute as an area.

**Generalization/Inheritance:** Generalization associations are used as a piece of class, segment, arrangement, and use case graphs to demonstrate that the tyke gets most of the characteristics, tasks, and associations that are characterized in the parent. Generalization indicates inheritance between classes.



Initial class diagram of city health care website.

Conclusion

While going through analysis. I have utilized use case diagram which associates with actor and framework. Likewise, functional and non-functional requirements are delineated. By the utilization of Moscow prioritization, we take the fundamental significant and essential prerequisites in this venture investigation. At long last, structure is set apart by the example called MVC and as required beginning class diagram is given. Finally, with the majority of this above project analysis is effectively finished.