

## Assignment 1

<1> What are agile Process and explain any one of them.

Ans) Agile Software engineering combines a philosophy and a set of development guidelines.

The Philosophy encourages customer satisfaction, and in early incremental delivery of software; small, highly motivated project teams; informal methods; minimal software engineering work products; and overall development simplicity.

The development guidelines stress delivery over analysis and design and active and continuous communication between developers and customers.

(2) Any agile Software Process is characterized in a manner that address a number of key assumptions about the majority of software projects:

i) It is difficult to predict in advance which software requirements will persist and which will change. It is equally difficult to predict how customer priorities will change as the project proceeds.

ii) For many types of software, design and construction are interleaved. That is, both activities should be performed in tandem so that design models are proven as they created. It is difficult to predict how much design is necessary before construction is used to prove the design.

(iii) Analysis, design, construction, and testing are not as predictable from a planning point of view as we might like.

③ Given these three assumptions, an important question arises: How do we create a process that can manage unpredictability? Answer is adaptability. An agile process, therefore, must be adaptable.

④ An agile software process must adapt incrementally for forward progress.

⑤ To accomplish incremental adaptation, an agile team requires customer feedback.

⑥ An effective catalyst for customer feedback is an operational prototype or a portion of an operational system.

⑦ Iterative approach enables the customer to evaluate the software increment regularly, provide necessary feed-back to the software team, and influence the process adaptations that are made to accommodate the feed back.

⑧ Common Agile Process models: XP, TXP, Scrum, Agile modeling and Agile unified process.

⑨ XP Process:-

① Extreme Programming (XP) uses an object-oriented approach as its preferred development paradigm & encompasses a set of rules & practices that occur within the context of four frame-work activities: Planning, design, coding and

testing. Idey XP activities.

- ① Planning:- ① The planning activity begins with listening - a requirements gathering activity that enables the technical members of the XP team to understand the business context for the software and to get a broad feel for required output and major features and functionality. Requirement in XP is known as user story.  
② The highest <sup>value</sup> and riskiest stories will be released in early increment.  
③ After delivery of 1<sup>st</sup> increment project velocity is calculated which helps for production & delivery of future increment.

- ② Design:- ① XP design rigorously follows the KIS (Keep It Simple) principle.  
② A simple design is always preferred over a more complex representation.  
③ The design provides implementation guidance for a Story as it is written - nothing less, nothing more.  
④ The design of extra functionality is discouraged.

- ③ Coding:- ① Before coding, unit tests are developed that will exercise each of the Stories that is to be included in current release.  
② Depending on unit test developer focuses on what is to be implemented to pass the test.

- ③ Once the code is complete, it can be unit-tested immediately, thereby providing instantaneous feed back to the developers.
- ④ XP encourages pair programming.

(iv) Testing :- Since, individual unit tests are organized into a "universal testing suite", integration and validation testing of the system can occur on a daily basis.

⑤ XP acceptance tests, also called customer tests, are specified by the customer and focus on overall system features and functionality that are visible and reviewable by the customer.

## 2) Difference between waterfall and spiral model.

### Ans) Waterfall model

① Waterfall model works in sequential method.

② In waterfall model errors or risk are identified and rectified after the completion of stages.

③ Waterfall model is applicable for small and simple project.

④ No concept of incremental delivery.

⑤ Flexibility to adapt is difficult & once the same activity is completed there is no going back.

⑥ High risk of failure.

⑦ Cost of applying is comparatively low.

⑧ Needs a thorough understanding of requirements from the beginning.

### Spiral model

① Spiral model works in combination of iteration and sequential method i.e. Evolutionary method.

② In spiral model errors or risks are identified and rectified earlier due to its iterative pattern.

③ Spiral model is applicable for large and complex project.

④ Encourage incremental delivery after one circuit is completed.

⑤ Flexibility to adapt is easy due to its iterative nature and same work activity can be repeated.

⑥ Risk of failure is low.

⑦ Spiral model is very expensive and not applicable for tight budget.

⑧ Requirements can be added up in the new iterations if required.

b) What are current trends in IP and explain any two of them.

- Ans) Current Trends in I.P. :-
- 1) Internet of Things (I.O.T)
  - 2) Blockchain
  - 3) Artificial Intelligence (A.I)
  - 4) Virtual Reality
  - 5) Augmented Reality
  - 6) Low-Code development
  - 7) Human Augmentation
  - 8) Mixed Reality (M.R)
  - 9) Cloud Computing

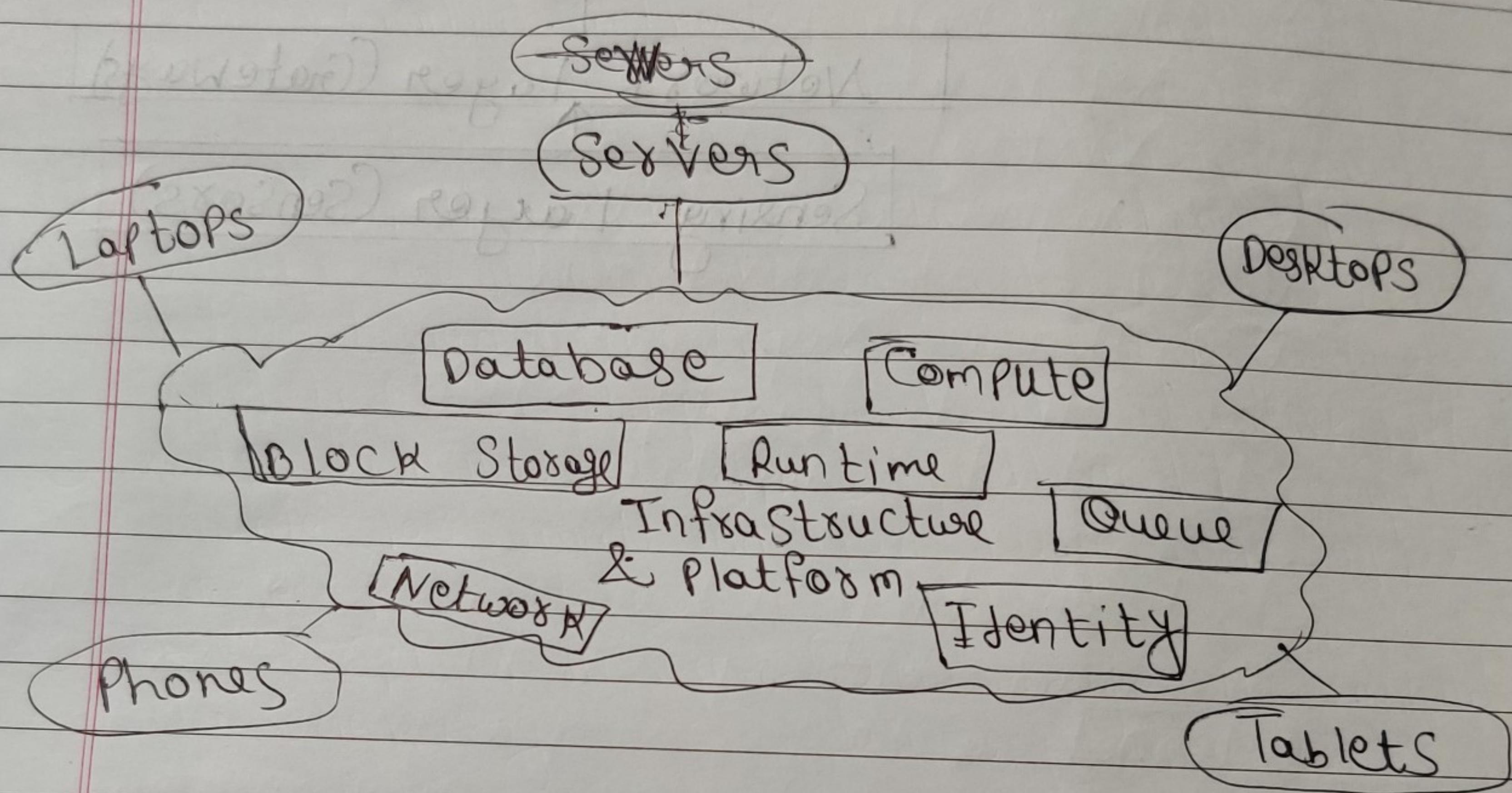
Cloud Computing:- Cloud Computing is the on-demand delivery of IT resources over the internet with Pay-as-you-go pricing.

(i) Instead of buying, owning and maintaining physical data centers and servers, you can access technology services such as computing power, storage, and databases, on an as-needed basis from a cloud provider like Amazon Web Services (AWS).

(ii) Application:- Organizations of every type, size and industry are using the cloud for a wide variety of use cases, such as data backup, disaster recovery, email, virtual desktops, software development and testing, big data analytics and customer-facing web applications.

ain  
 (iv) Characteristics:- Agility, Elasticity (amount of resources can be scaled), cost savings and Deploy globally in minutes.

(v) Architecture:-



Internet of Things (IoT):- i) IoT is the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment.

ii) Applications:- RFID to track objects, wearables like smart watch to monitor calorie expenditure and heartbeats, GPS tracking belts, Smart grid and energy saving, etc.

iii) Characteristics:- Connectivity, Intelligence and Identity, Scalability, Dynamic and self adapting, Architecture & Safety.

(iv)

Architecture:-Application Layer (Application)Data Processing Layer (Processing unit)Network Layer (Gateways)Sensing Layer (Sensors)

Q.4) Develop a Software requirement Specification (SRS) for developing a software for Hospital Management System.

### 1.) Introduction:-

<<1.1>> Purpose :- The main purpose of our system is to make hospital task easy and is to develop Software that replaces the manual hospital System into automated hospital management System. This document serves as the unambiguous guide for the developers of this Software System.

### 1.2) Document Conventions:-

- i) HMS - Hospital Management System
- ii) GUI - Graphical User Interface
- iii) PHID - Patient Hospital Identification Number

### 1.3) Scope of The Project:-

- i) The purpose of this specification is to document requirements for a system to manage the hospital.
- ii) The specification identifies what such a system is required to do.
- iii) The H.M.S will manage a waiting list of patients required different treatments.
- iv) The availability of beds will be determined and if beds are available the next appropriate patient on the list will be notified.
- v) Nurses will be allocated to wards depending on ward size, what type of nursing is needed, operating schedules, etc.

- (vi) The HMS will allow hospital administrative staff to access relevant information efficiently and effectively.
- (vii) The goal of HMS is to manage nurses, patients, beds, and patients medical information in an cost-effective manner.

## 2) Overall Description:-

### 2.1) Product Perspective:-

The various system tools that have been used in developing both the frontend, backend and other tools of the project are being discussed in this section.

i) Frontend:- JSP, HTML, CSS, Java Script are utilized to implement the frontend.

ii) Backend:- Database is handle using MySQL

iii) API : JAVA, PHP are used for connecting Frontend & backend.

### 2.2) Product Functions:-

i) The HMS will allow the user to manage information about Patients, nurses, & beds. Patient management will include the checking-in and checking-out of patients to & from the hospital.

ii) The HMS will also support the automatic backup & protection of data.

## 2.3) Operating Environment:-

Minimum requirements for smooth functioning of Software:-

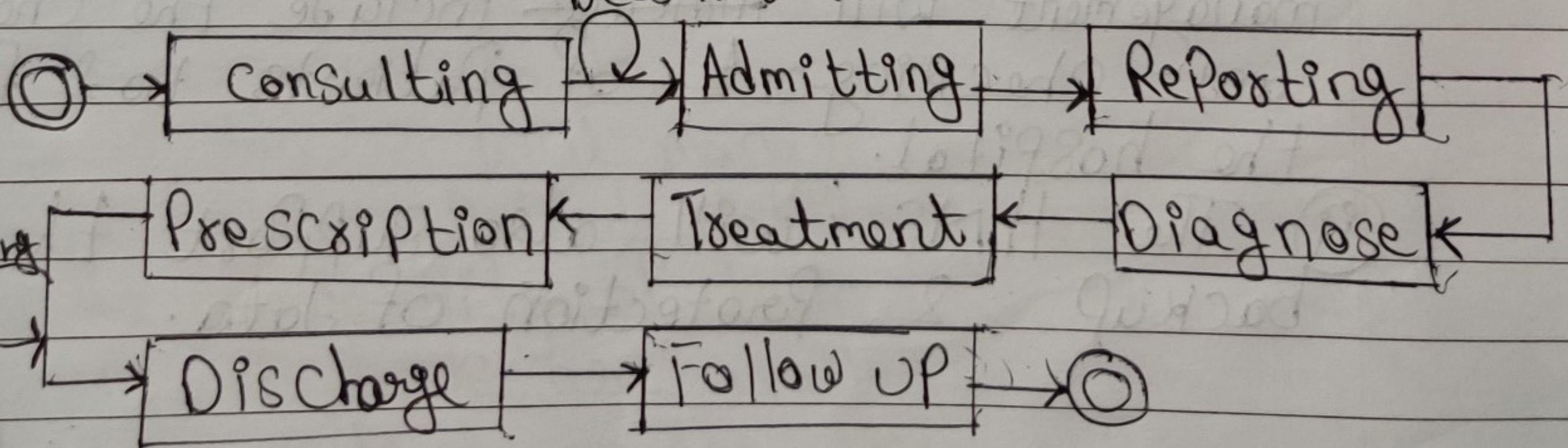
- i Processor: i3 or higher (2 GHz+)
- ii RAM: 4 GB+
- iii Disk Space: 20 GB+
- iv OS: Windows 10/8 or MAC OS.

## 2.4) Design & Implementation Constraints:-

- i GUI only in English.
- ii No guest users are allowed.
- iii Minimum information <sup>limit</sup> should be enough for creating account.

## 2.5) Assumption and Dependencies:-

- i Staff is familiar with modern technology.
- ii Minimum 100 computers are available.
- iii Maximum no. of users will always be less than 10,000.



### 3.1) External Interface Requirements:-

#### 3.1.1) User Interfaces:

System is GUI based. Keyboard & mouse can be used to interact with software.

3.2) and 3.3) Hardware & Software Interface already mentioned in operating environment & product perspective.

#### 4.) System Features:

##### 4.1) System Features:-

i) Working Schedule: Assigning nurses to doctors & doctors to patients.

ii) Admissions: Admitting Patients, assigning the patients to appropriate wards.

iii) Patient Care: Monitoring Patients while they are in the Hospital.

iv) Surgery Management: Planning and organizing the work that Surgeons & Nurses perform in the operating rooms.

v) Ward Management: Planning and coordinating the management of wards & rooms.

vi) Waiting List: Monitoring to see if there are any patients waiting for available beds, assigning them to doctors and beds once these become available.

## 5) Other Non-Functional Requirements:-

### 5.1) Performance Requirements:-

The Performance of Software is at its best when the following regularly are done:-

- i) Password Management.
- ii) Regular Database Archiving.
- iii) Virus Protection.

### 5.2) Safety Requirements:-

Human are error-prone, but the negative effects of common errors should be limit. e.g.: - users should realize that a given command will delete data, and be asked to confirm their intent or have option to undo.

### 5.3) Software Requirements:-

i) Each member is required to enter an individual Username and Password when accessing the software. Administrators have the option of increasing level of password security their members use.

ii) The data in the database is secured through multiple layers of protection.

iii) One of those security layers involves member Passwords. For maximum security of your software, each member must protect their password.

## 5.4 Software Quality Attributes

The Quality of the system is mentioned maintained in such a way that it can be very user-friendly. The Software Quality attributes are assumed as follows:-

- i) Accurate and hence reliable.
- ii) Secured.
- iii) Fast Speed.
- iv) Compatibility.