Chapter 5

System Analysis

**5.1 INTRODUCTION**

Software analysis and design include all activities, which help the transformation of requirement specification into implementation. Requirement specifications specify all functional and non-functional expectations from the software. These requirement specifications come in the shape of human-readable and understandable documents, with which a computer has nothing to do.

Software analysis and design is the intermediate stage, which helps human-readable requirements to be transformed into actual code.

requirements, available time, human resources, and financial requirements.

**Data Flow Diagram:**

The data flow diagram is a graphical representation of the flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow, and stored data. The DFD does not mention anything about how data flows through the system. There is a prominent difference between DFD and Flowchart. The flowchart depicts the flow of control in program modules. DFDs depict the flow of data in the system at various levels. DFD does not contain any control or branch elements.

**Types of DFD:**

Data Flow Diagrams are either Logical or Physical.

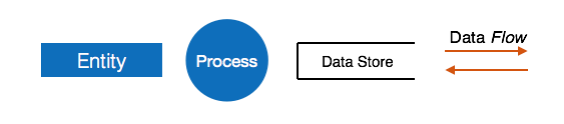
• **Logical DFD:**

This type of DFD concentrates on the system process, and flow of data in the system. For example in a Banking software system, how data is moved between different entities.

• **Physical DFD:**

This type of DFD shows how the data flow is implemented in the system. It is more specific and close to the implementation.

**DFD Components**

DFD can represent the Source, destination, storage, and flow of data using the following set of components

**• Entities:**

Entities are the source and destination of information data. Entities are represented by a rectangle with their respective names.

**• Process:**

Activities and actions taken on the data are represented by circles or Round-edged rectangles.

**• Data Storage:**

There are two variants of data storage - it can either be represented as a rectangle with an absence of both smaller sides or as an open-sided rectangle with only one side missing.

**• Data Flow:**

The movement of data is shown by pointed arrows. Data movement is shown from the base of the arrow as its source towards the head of the arrow as the destination.

**5.2 System Analysis**

**5.2.1 Introduction to System Planning**

Before starting a software-hardware project, it is essential to determine the tasks to be performed and properly manage the allocation of tasks among individuals involved in the software development. Hence, planning is important as it results in effective software development. Project planning is an organized and integrated management process, which focuses on activities required for the successful completion of the project. It prevents obstacles that arise in the project such as changes in projects or organization's objectives, non-availability of resources, and so on. Project planning also helps in better utilization of resources and optimal usage of the allotted time for a project. The other objectives of project planning are listed below.

• It defines the roles and responsibilities of the project management team members.

• It ensures that the project management team works according to the healthcare/covid objectives.

• It checks the feasibility of the schedule and user requirements.

• It determines project constraints. Several individuals help in planning the project. These include senior management and the project management team. Senior management is responsible for employing team members and 29 providing resources required for the project. The project management team, which generally includes project managers and developers, is responsible for planning, determining, and tracking the activities of the project. Project planning should be effective so that the project begins with well-defined tasks. Effective project planning helps to minimize the additional costs incurred on the project while it is in progress. For effective project planning, some principles are followed. These principles are listed below.

• Planning is necessary: Planning should be done before a project begins. For effective planning, objectives and schedules should be clear and understandable.

• Risk analysis: Before starting the project, senior management and the project management team should consider the risks that may affect the project. For example, the user may desire changes in requirements while the project is in progress. In such a case, the estimation of time and cost should be done according to those requirements (new requirements).

• Tracking of project plan: Once the project plan is prepared, it should be tracked and modified accordingly.

• Meet quality standards and produce quality deliverables: The project plan should identify processes by which the project management team can ensure quality in software. Based on the process selected for ensuring quality, the time and cost for the project are estimated.

• Description of flexibility to accommodate changes: The result of project planning is recorded in the form of a project plan, which should allow new changes to be accommodated when the project is in progress.

Software design approach

Here are two general approaches for software design:

**Top-Down Design**

We know that a system is composed of more than one subsystem and it contains several components. Further, these sub-systems and components may have their onset of sub-system and components and create hierarchical structure in the system. 30 Top-down design takes the whole software system as one entity and then decomposes it to achieve more than one subsystem or component based on some characteristics. Each subsystem or component is then treated as a system and decomposed further. This process keeps on running until the lowest level of the system in the top-down hierarchy is achieved. Top-down design starts with a generalized model of the system and keeps on defining the more specific part of it. When all components are composed the whole system comes into existence. Top-down design is more suitable when the software solution needs to be designed from scratch and specific details are unknown.

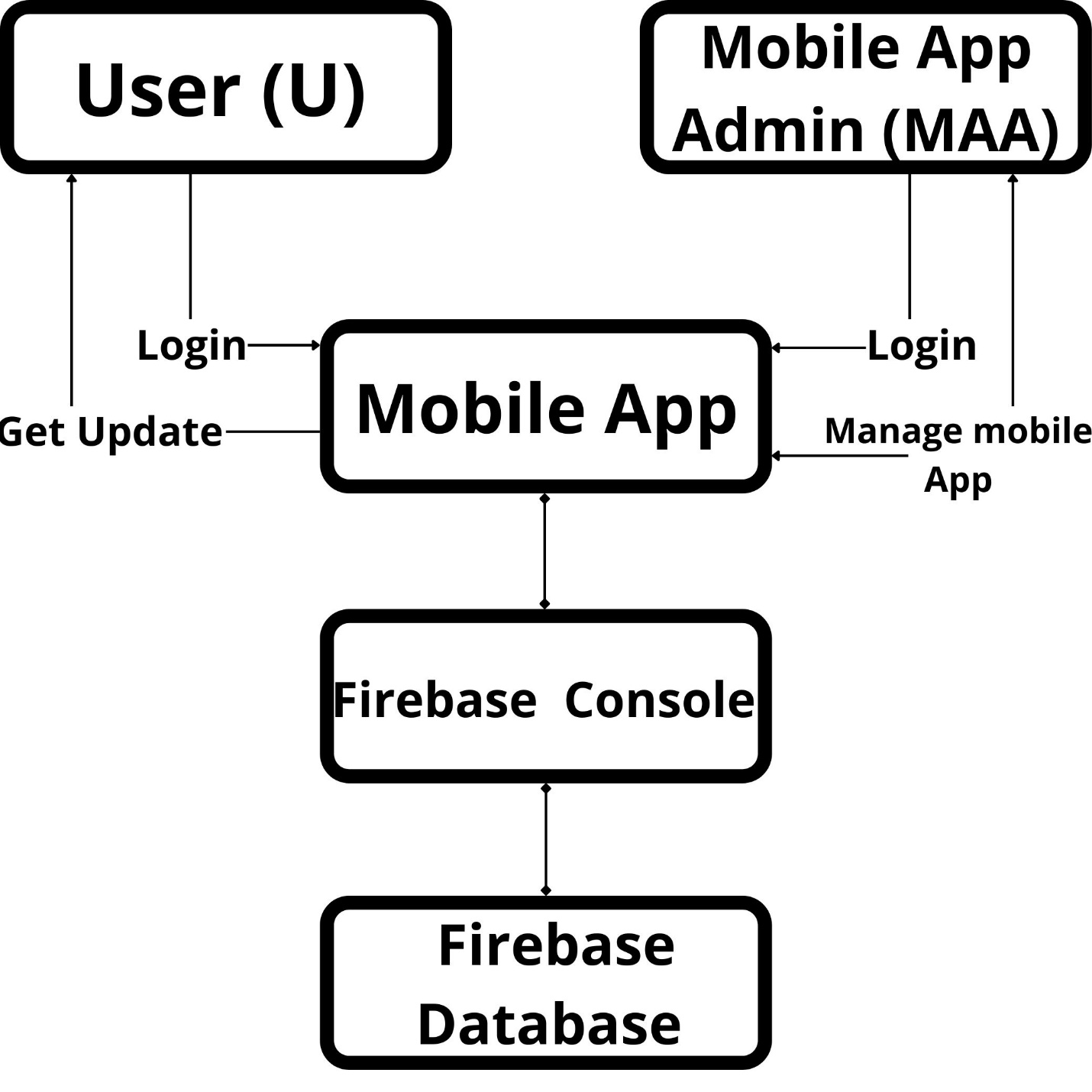
**Bottom-up Design**

The bottom-up design model starts with the most specific and basic components. It proceeds with composing a higher level of components by using basic or lower-level components. It keeps creating higher-level components until the desired system is not evolved as one single component. With each higher level, the amount of abstraction is increased. A bottom-up strategy is more suitable when a system needs to be created from some existing system, where the basic primitives can be used in the newer system.

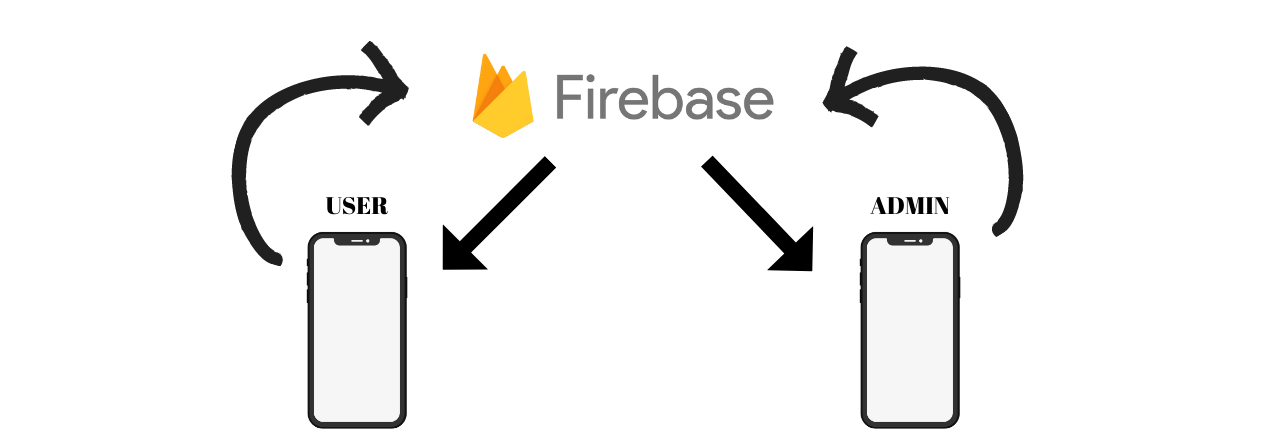
**5.2.2 Software Design Approach we chose:**

Bottom-up Design We had decided to take the modular way to develop our project and the bottom-up design approach was best suited for our project development. Firstly, we developed the core components of the Firebase Database and then we kept developing the rest of the components and continuously combined them with their respectively dependent components.

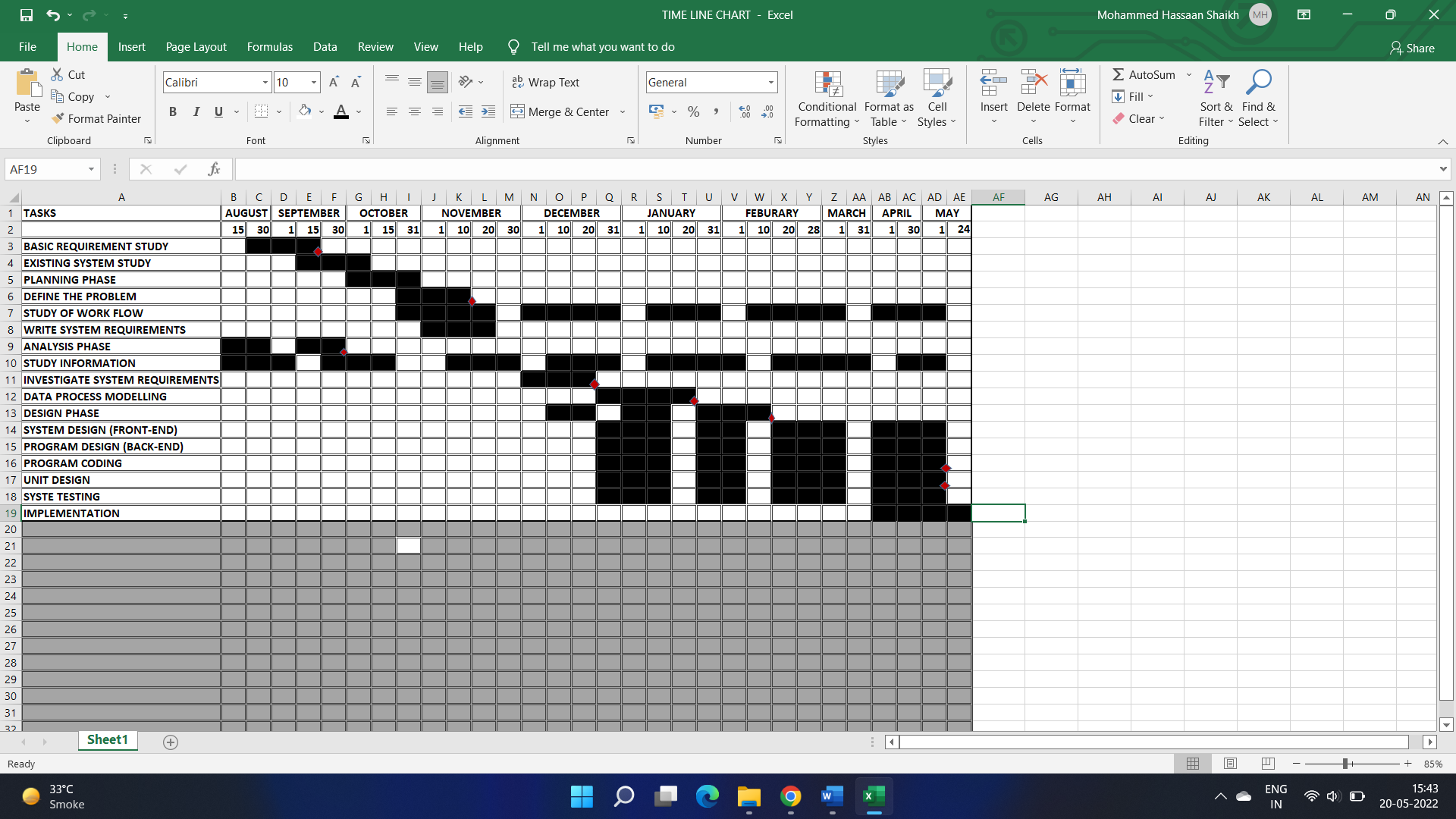
**5.3 Block Diagram**



**5.4 System Architecture**



**5.5 Timeline Chart**



**5.6 Cost Estimation**

**Total number of weeks:** 5th Sem + 6th Sem

=13+16 weeks

=29weeks

**Number of hours per week:** 9 hours

**Total number of Hours:** 29\*9

**=** 261 Hours

**Cost of development per Hour:** 80 INR

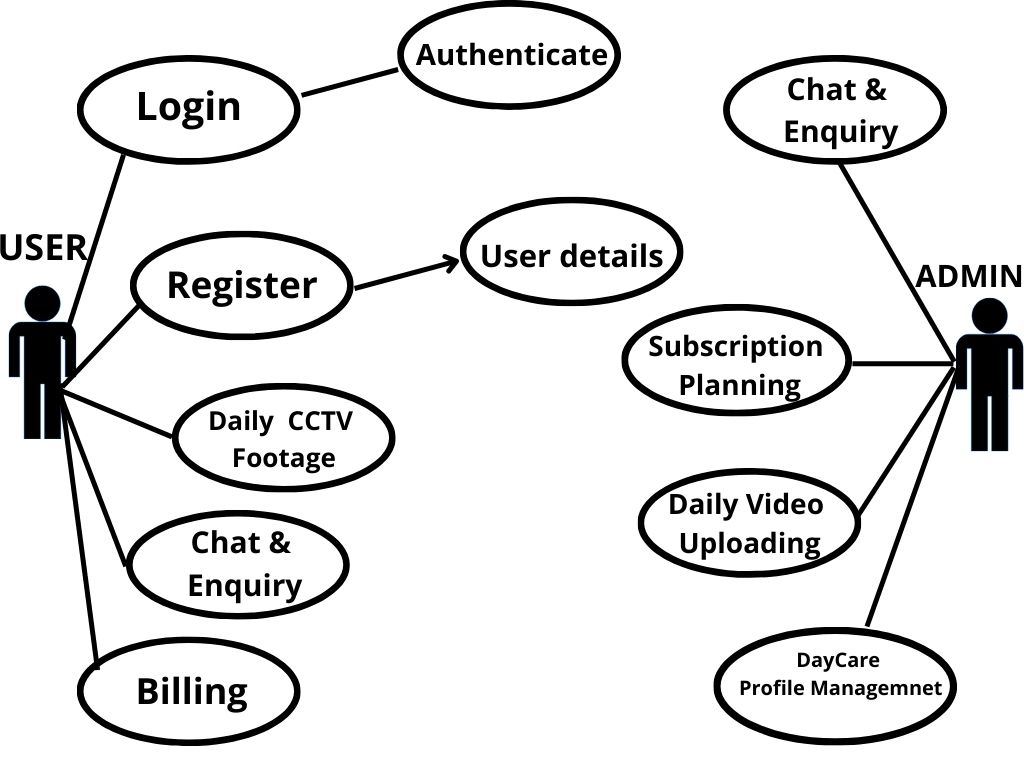
**Total Cost:** Total no. of Hours \* Cost of Development per Hour

=261 \* 80

= **20,880 INR**

**5.7 UML Diagram**

**Use Case Diagram:**

****

**5.8 Feasibility Study**

A Feasibility Study is an analysis of how successfully a project can be completed, accounting for factors that affect it such as economic, technological, legal, and scheduling factors. Project managers use feasibility studies to determine potential positive and negative outcomes of a project before investing a considerable amount of time and money into it. There are three types of feasibility, they are defined as follows:

1. **Technical Feasibility**

It has been determined that the technology needed for the proposed system is available and that this technology can be integrated into the application. Technical evaluation has also evaluated the existing system to find that it cannot be upgraded in keeping with the user’s needs. Hence, we need to create an entire system that caters to the specific needs of the user. The end-user can be equipped with the pre-mentioned hardware and software requirements.

**Hardware Requirements**

* Ram: 256 MB or more
* Processor: 2nd generation Intel Core or newer, or AMD CPU
* HDD: 10 GB or more
* Android (Basic)

1. **Economical Feasibility**

The economic feasibility of the system is mainly concerned with its financial aspects. It determines whether the project is economically feasible. As the hardware and software are already available easily in the market, no further investment is to be made in that direction, the only cost involved is that of implementing the system. The entire system is made in Android studio hence no costs are required in that direction however hosting services will begin charging after a certain period.

**It was decided that the project was technically feasible because of the following:**

* Necessary technology exists to do what is suggested.
* The system would be expendable if so decided.
* The system can be integrated with other networking tools and the output of the system can be provided to other tools for high-end analysis.
* This system is technically more secure.

1. **Operational Feasibility**

Operational feasibility is a measure of how well a proposed system solves the identified problems and takes advantage of the opportunities identified in the scope of research. The system should also satisfy the requirements identified in the requirement analysis phase of system development. Since the software is intended in making the encryption process customized, The user must have the fundamental knowledge of the computer(he must know how to do character selection and it is assumed that the user will have the basic skills at the system has its application at a business level). The proposed system is intended at providing a high level of abstraction(AES algorithm is applied just at one click and no coding is shown) to the user so that even any person with average knowledge in the working of a computer is able to use it efficiently. Since all these requirements are easy and affordable, the customized system is operationally feasible.

**4. Scheduling Feasibility**

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization

estimates how much time the project will take to complete