Methodology

To develop a software product, there must be a clear understanding among team representative about when and what to do. Software development life cycle plays the most important role in developing a software. Software life cycle model is a pictorial and diagrammatic representation of the software life cycle. A life cycle model represents all the methods required to make a software product transit through its life cycle stages. It also captures the structure in which these methods are to be undertaken. There are different software models to implement the SDLC like waterfall model, spiral model, incremental model, iterative model, v-model, agile model, rad model etc.

Among these, we have implemented waterfall model in our project. Waterfall model was introduced by Winston Royce in 1970 which was the first process model to be introduced. Waterfall model is the basic software development life cycle model which is very simple but idealistic. It is a systematic and sequential approach to software development that begins with projects specification of requirements and progresses through planning, modeling, construction and deployment, culminating in ongoing support of the completed software. In this model, each phase must be completed before the beginning of the next phase so that there is no overlapping of the phases. Typically, the output of the one phase acts as the input for the next phase sequentially.

The phases of waterfall model are explained below:

1. Requirements definition:

The aim of this phase is to understand the exact requirement of the project and document them properly. Firstly, all the requirements of the customer regarding software are gathered and analyzed which are then documented in a software requirement specification (SRS) document.

1. System and software design:

The system design process allocates the requirements to either hardware or software systems. It establishes an overall system architecture.

1. Implementation and unit testing:

During this stage, software design is converted into a source code using any suitable programming language where each designed module is coded. Unit testing involves verifying that each unit meets its specification.

1. Integration and system testing:

In this stage, the integration of different designed module in the form of code are integrated and tested as a complete system in order to ensure that the full working system is obtained.

1. Operation and Maintenance

This is the final stage where the system is installed and brought into practical use. Maintenance involves correcting errors that were not discovered in earlier stages of the life cycle in order to enhance and improve the system services. Corrective maintenance, perfective maintenance and adaptive maintenance are carried out in this stage.

The different sequential phases of the waterfall model are shown in the figure below:

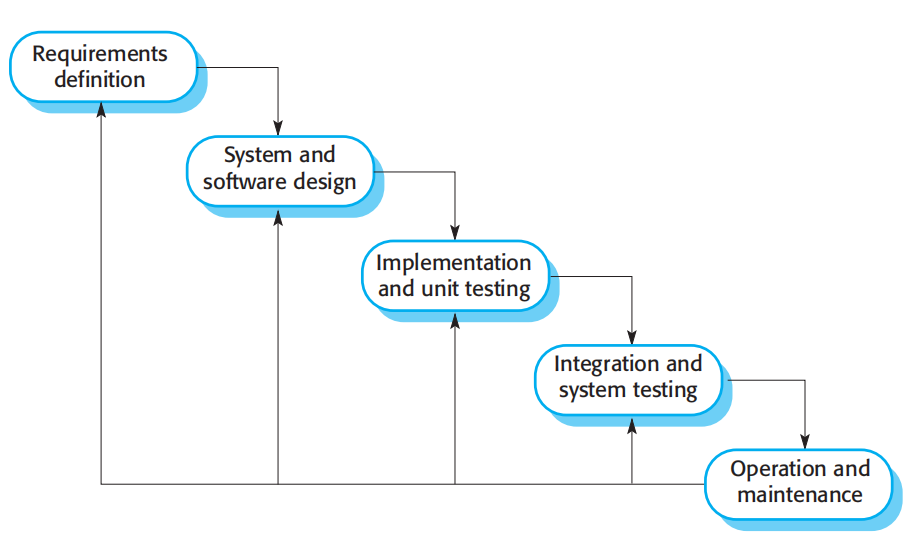


Fig: Waterfall Model

Why Waterfall Model?

We have decided to use this model as it is very simple and easy to process each phase one at a time. This model has very clear and well-understood milestones where process, actions and results are very well documented. This model works well for projects like ours where work is divided among each members and each member is assigned with phase which are well understood. Here, the requirements, tools and technology being used in this project is consistent and constant which results in the effective implementation of this model. As the start and end points of each phase are fixed, it make us easier to cover the progress.

Requirement Engineering

Feasibility study

Feasibility study is an analysis and evaluation of a proposed project to ensure that it is technically, economically and operationally feasible. It mainly focus on whether the proposed project idea should be proceed or not in terms if various factors.

Economic feasibility

We need to ensure that development of project should be economically feasible. As it is a light basic project, the only requirement of this project is a working computer system with some required software components and development platforms installed. Any mid spec computer is capable of installing and using the end product of this project. Thus, this project is economically feasible.

Technical feasibility

Technical feasibility evaluates the proposed project can compiles with current technologies, which are needed to accomplish needed requirements. The web application of end project is supported by almost all devices with minimum hardware and software requirements. Thus, this project is technically feasible.

Operational feasibility

The proposed project provides the application of cryptography in communication process so that the end-to-end encryption is applied during communication. Technically speaking, it’s a web application that provide secure exchange of information and data in this world of the internet.

Schedule feasibility

Here we analyzed the the time required to complete the project and identified that the proposed project will fail if the time frame of software development exceeds the proposed time frame.

Gantt Chart

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S.N | Weeks/Phases of  Project | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1. | Feasibility Study |  |  |  |  |  |  |  |  |  |
| 2. | Requirement gathering and  analysis |  |  |  |  |  |  |  |  |  |
| 3. | System design and  Coding |  |  |  |  |  |  |  |  |  |
| 4. | Unit testing,  Integration and  System testing |  |  |  |  |  |  |  |  |  |
| 5. | Documentation |  |  |  |  |  |  |  |  |  |

Requirement analysis

Hardware requirement

* For developer
* Processor: Intel Pentium 4 or above
* RAM: 2 GB or above
* Available disk space: 5GB+
* Running internet connection
* A server

Software requirement

* Any operating system
* Language: Python
* Text editor

Implementation/Coding

Tools and Technique used:

* Anaconda framework