Online NITK Website and Management Software

Project Report

Submitted in partial fulfilment of the requirements for the degree of

MASTER OF TECHNOLOGY in

COMPUTER SCIENCE AND ENGINEERING

by

Dayma Khan (212CS006)

Lokesh Dohare (212CS011)

Nikita Patil (212CS015)

Pallavi Gupta (212CS016)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE -575025

January, 2021

1. Feasibility Study

Feasibility study is carried out based on many purposes to analyze whether software products will be right in terms of development, implantation, contribution of project to the organization etc. This contains an analysis that considers all of the software's important aspects, including economic, technical, and scheduling issues, to determine the project's chances of success. For the software's business development, a feasibility study is required.

1.1. Financial Feasibility

It has no cost because it is a web application. All of the frameworks and APIs that were used are open source. Potential clients will not be charged anything. It is clear from this that the idea is financially viable.

1.2. Technical Feasibility

This is a desktop application that has been developed using a variety of frameworks, libraries, and APIs that are publicly available. As a result, the project is doable from a technological standpoint.

1.2.1 Software Requirements

- Browser : Google Chrome, Firefox, etc
- Code Editor : Visual Studio Code, Sublime Text, etc.
- Technology Stack :
    - Frontend : HTML, CSS, Javascript, Bootstrap
    - Backend : Python Django Framework
    - Database : SQLite

1.2.2. Hardware Requirements

- Intel i3 processor or higher and minimum of 4GB RAM for project.

1.3. Schedule Feasibility

Date/Week	Project Plan
2-12-2021	Assignment 4 issued
Week 1	Requirement Analysis and Design
Week 2	Frontend Development

Week 3	Backend Development
Week 4	Integrating frontend and backend
Week 5	Testing and Documentation

2. Software Development Life Cycle

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

In this project, we have used the **Waterfall Model**.

The reason for selecting this model is that it is a great model for small-sized applications which do not need multiple rounds of software development cycles. Also, the product requirements are stable and well-understood at the very beginning of the project. The Waterfall model is very easy to manage due to the rigidity of the model, which suits our purpose.

The different stages of the waterfall model are as follows:

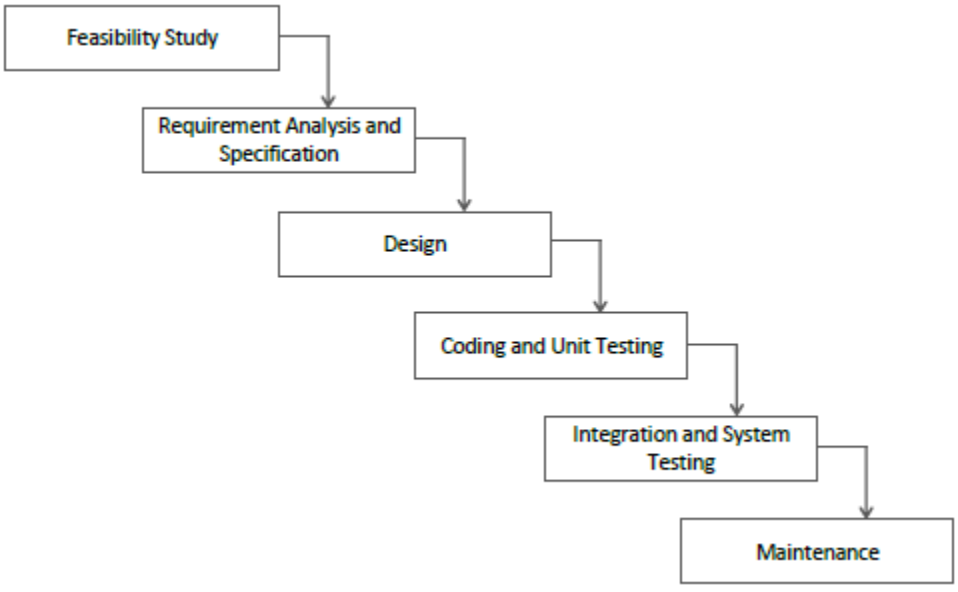


Image Source : https://www.geektonight.com/classical-waterfall-model-software-engineering/

2.1. Feasibility Study

- The main aim of the feasibility study is to determine whether it would be technically and financially feasible to develop the product.
- The feasibility study activity involves analysis of the problem and collection of all relevant information relating to the product.
- It was carried out in week 1.

2.2. Requirement Analysis and Specification

- The goal of the requirements gathering activity is to collect all relevant information from the customer regarding the product to be developed. Once the requirements have been gathered, the analysis activity is taken up.
- All possible requirements of the system to be developed are captured in this phase and documented in a document.
- Carried out in weeks 1 with information from https://www.nitk.ac.in/

2.3. Design

- During the design phase, the software architecture is derived from the requirements gathered in the previous phase.
- This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- Carried out in week 2.
- Directory Structure :

  The Main Folder Contains subfolders for project and app, readme.md file. Within app folder, we have 3 folders : static - containing images, scripts and stylesheets; templates - containing HTML pages pycache : containing python logs; and files like views.py, urls.py, admin.py etc. Project folder contains 5 files : init.py : __init__.py, asgi.py, settings.py, urls.py, wsgi.py. Also, there is a database file db.sqlite3.

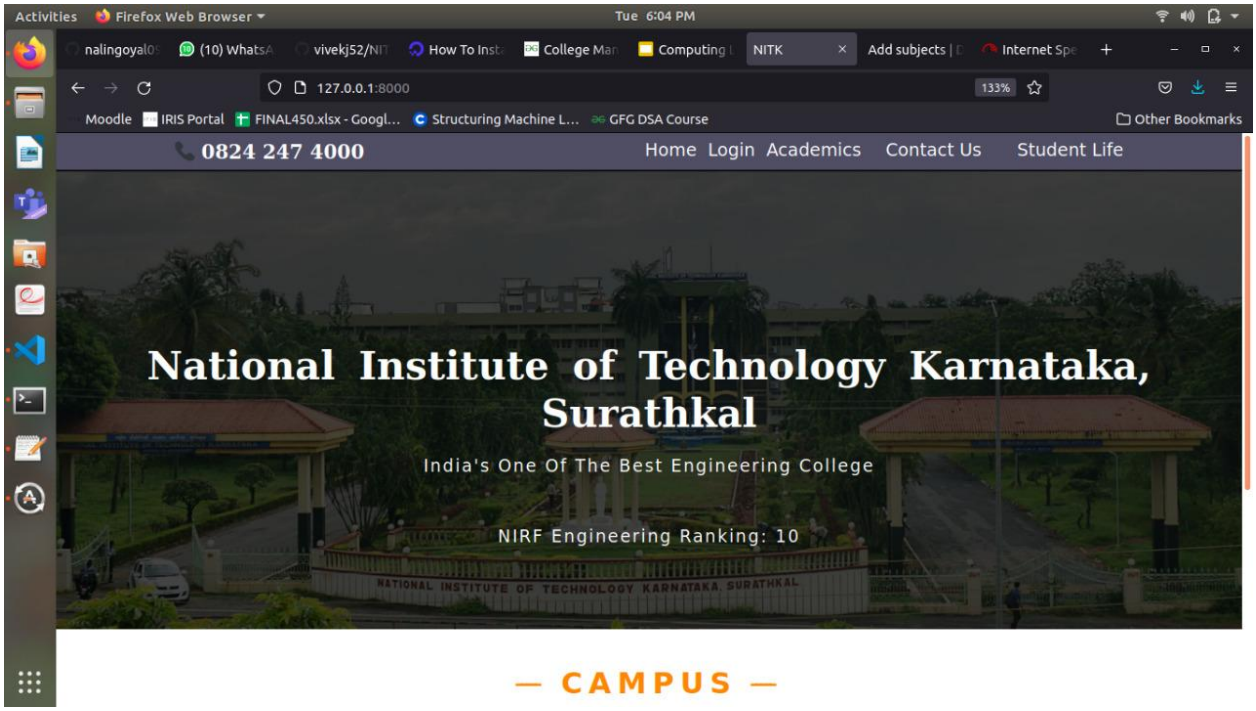
2.3.1. UML Design

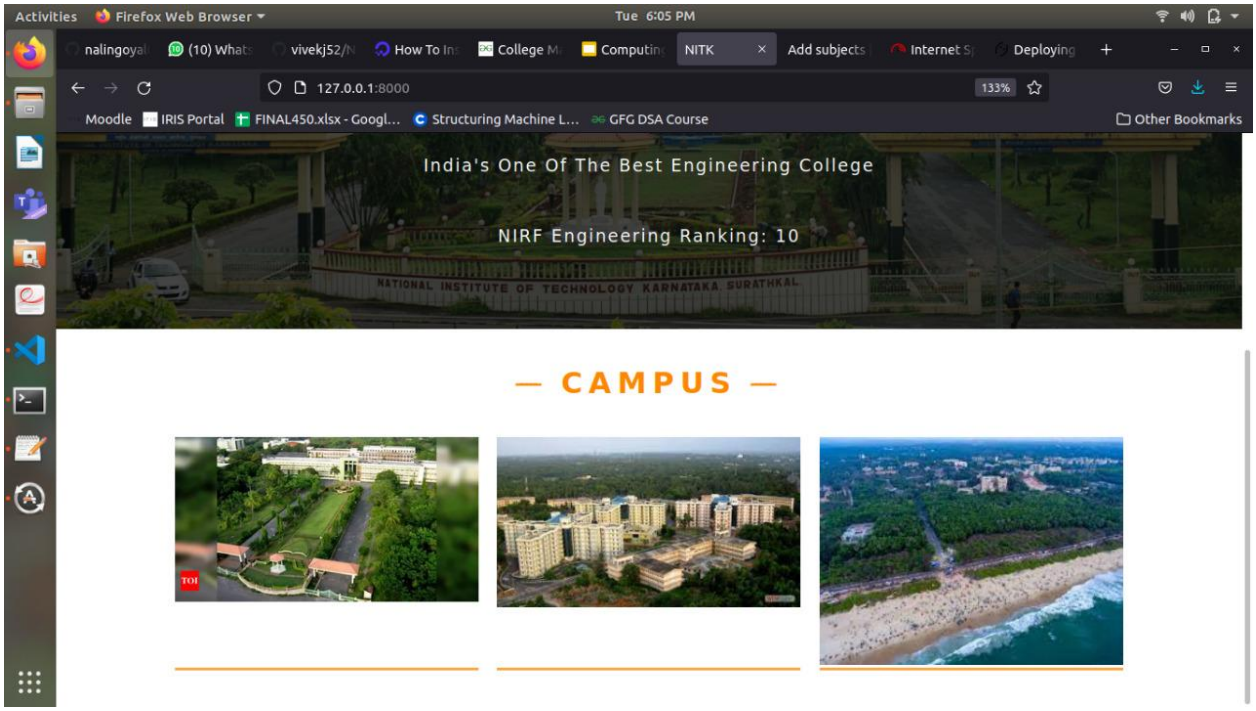
The Unified Modeling Lanage (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system. Below is the Use Case Diagram of our system :

2.4. Coding and Unit Testing

- The purpose of the coding and unit testing phase of software development is to translate the software design into the source code.
- With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase.
- Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- We implement the design in week 2 and 3.
- The frontend is developed using HTML, CSS, Javascript and Bootstrap.
- The system backend is developed using Django which is a Python-based free and open-source web framework that follows the model–template–views architectural pattern.
- Django documentation : https://docs.djangoproject.com/en/4.0/
- The system database is built using SQLAlchemy is the Python SQL toolkit and Object Relational Mapper that gives application developers the full power and flexibility of SQL.

- SQLite was chosen as the database as it suits well to the relational model of our application.
- SQLite Documentation : https://docs.python.org/3/library/sqlite3.html





2.5. Integration and System Testing

- During this phase, all the modules are integrated into a system in a planned manner.
- Integration of various modules are normally carried out incrementally over a number of steps. After each integration step, the partially integrated system is tested.
- Testing was simply carried out manually by passing valid and invalid data.
- Integration and testing was carried out during week 3 and 4.

2.6. Maintenance

- Maintenance involves monitoring and improving system performance, enhancing system services, and upgrading to newer versions.
- Some issues might come up in the client environment. To fix those issues, patches are released.