**VIT**Go

**Leave / Outpass Application**

By:

Bhavik Shah 17BCN7006

Aditya Parmar 17BCN7013

Arghya Biswas 18BCE7024

Gaurav S 18BCE7009

Shubham Mane 18BCD7029

1. **Introduction**
   1. Purpose
   2. Intended Audience
   3. Project Scope
   4. Operating Environment
   5. Design and Development Constraints
2. **System Features**
   1. System Features
3. **External Interface Requirements**
   1. User Interfaces
   2. Hardware Interfaces
   3. Software Interfaces
   4. Communication Interfaces
4. **Other Non-functional Requirements**
5. **Cost Complexity**
6. **Link to GitHub Repository to view the entire code**

# Introduction

Purpose

VITGo is a leave/outing automated system designed for VIT University - Andhra Pradesh. This system is an end to end module which enables a user (Student) to raise a request and an admin (Mentor) to approve / decline it. This is a robust system where Parent Verification, In-Out Time recording, Data Security have been taken care of. VITGo is built to be a secure, flexible, unique, transparent and user-friendly environment which aims to digitize the whole process thus removing fake paper trail.

Feasibility

The project VITGo has been undertaken after the feasibility study, which paves the way

deployment, phase development.

**Scope**

The VITGo is designed to run on the VIT server and to allow students to raise requests for their leave, trace the request status, modify them. On the Mentor Dashboard the software also allows the Mentor/ Mentor Coordinator to view requests, approve/decline requests. Whereas on the Hostel Dashboard the Warden/ Deputy Warden and Hostel Supervisors will be able to view and grant leave pass to the students. The data will be held in an Access database on the departmental server.

VITGo will provide ease to all the actors - students, mentors, hostel authorities and security

services in regard with leave/outpass sanctions and will ultimately eliminate the paperwork.

# Overall Product Description

Product Perspective

VITGo will provide a way in which existing paper-based work can be supplemented with

end-to-end robust leave management system. The system can be used independent of the

platform and device, be it on a smartphone, tablet or computer.

**Product Functionality**

The server will be responsible for storing each request generated, generating one time

passwords, generating QR Codes E-Pass for authorised requests, receiving and authenticating requests, generating statistics at the needs of each audits and maintaining and verifying security and user privacy. This server can also potentially contact all authorised students by email to give them username information, passwords, server address, OTP code, updates to the users from the Mentors/ Hostel Services etc.

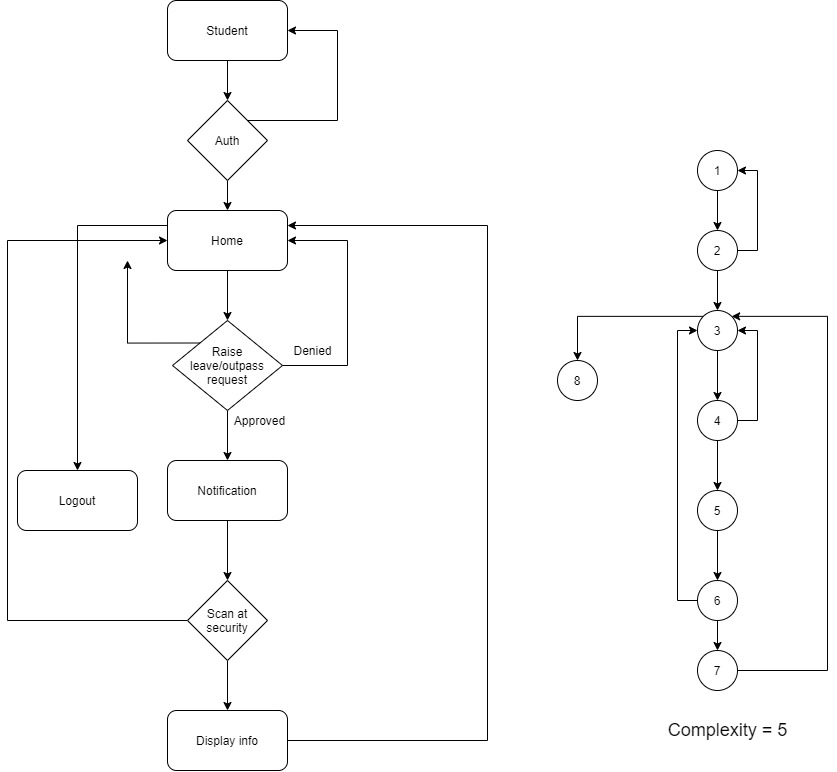
Process Flow – Student

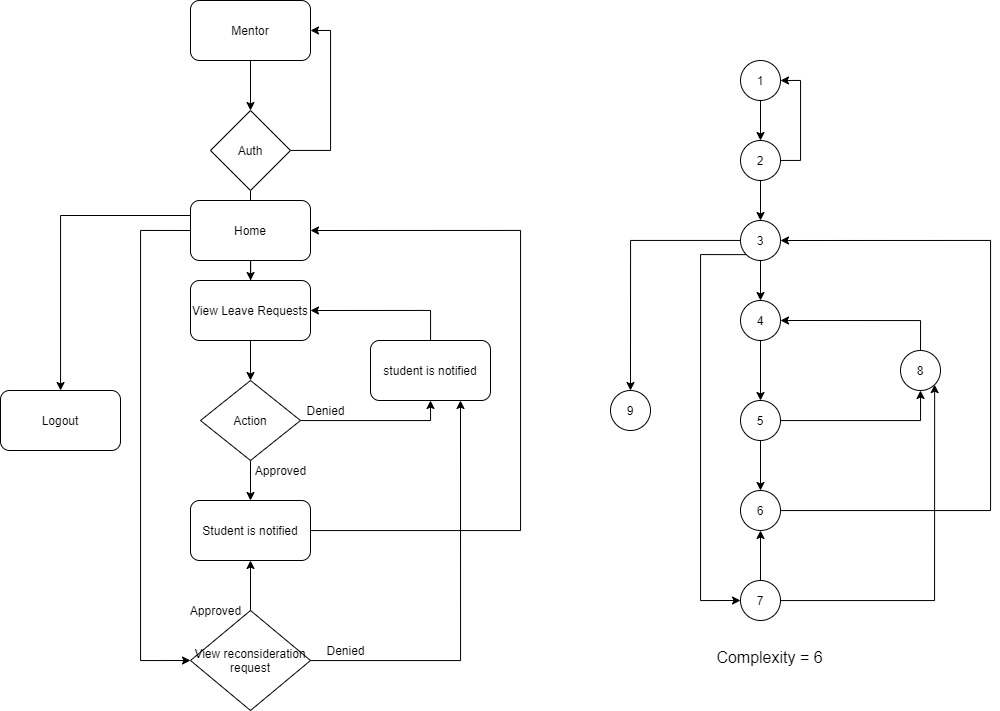
The Illustration diagram 1. depicts how the student raise the request and the activity which is

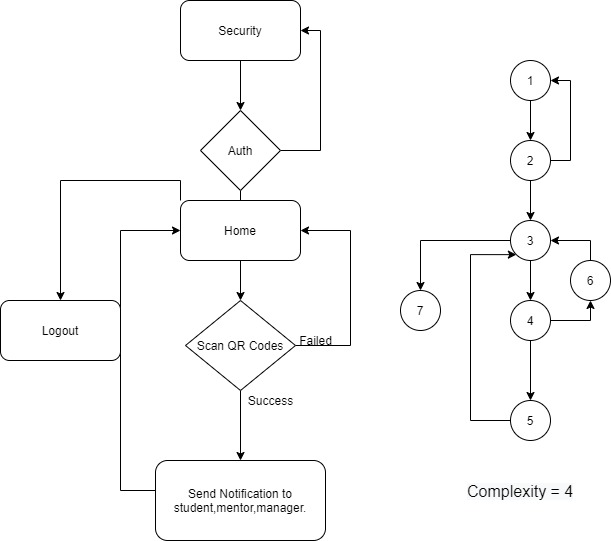
continued after the review from the mentor.

The Illustration diagram 2. depicts how the Mentor/Warden/Supervisor approves/declines the requests.

The Illustration diagram 3. depicts how the Security guard can verify the leave request the student displays.







Upper Classes and Characteristics

It is anticipated that three types of user will use the Licensed Software defined in this SRS.

1. IT staff/ Software Development Cells are expected to deploy and configure the System using the defined system interfaces. This will include running the whole system and maintenance of software after the handover and deployment of project.
2. The second type of user of the Hostel Warden/ Deputy Wardens / Hostel Supervisors and Managers who are expected to understand and use correctly the software interfaces defined with the appropriate design documentation.
3. Finally, it is expected that any student accorded with Hostel Services within the domain of Vellore Institute of Technology may access all of the leave request information such that the request is independently verifiable. This will include a web application presented using Hyper Text Markup Language (HTML) to allow a user to raise their request has been register under review includes the previous requests history.

Operating Environment

The VITGo software is directly made as a web application, so the computer hosting it must be capable of running HTML and should have internet. The system will be uploaded on the VIT server, in order to make it accessible for all the students, faculty mentors and wardens.

**Design and Implementation Constraints**

The VITGo Application provides end-to-end leave management system which copes with

malicious attacks provided certain constraints are met. Principally, all necessary steps should be taken to protect the VITGo System from unguarded attacks by using physical, network, storage and user security protection. These safeguards should be penetration tested by the SDC to ensure viability.

**User Documentation**

The users are the students or faculty/staff of Vellore Institute of Technology who are authorised by SDC, they will be able to raise/ view/ approve/ disapprove requests on the server. The application client will be available free of charge, and any purchase of the server software will be authorised to distribute it to their users.

**Assumptions and Dependencies**

This software in its initial phases of development depend on few third-party commercial

applications or any assumption. Student Development Team will take care of all the assumptions and dependencies. It will be the responsibility of SDC to either purchase/ develop the dependency as per the University IT norms.

# System Features

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Title of the system feature | Description | Priority | Functional Requirements |
| 1 | **Login** | This is used to login and  maintain the security by  authenticating the users | 9 | 1. Should accept the user  name password  2. A case insensitive  comparison is done for a  user name and a case  sensitive comparison is  done for a password  3. If the correct user id and  password are supplied  then, Main Menu should  be displayed  4. If an invalid user id or  password is entered then  the system should display  error message “Invalid ID  or password” and should  quit the application.  5. Username -  Students - Registration Id  Faculty - Employee Id  Staff - Employee Id  6. VTop Login Credentials  can be used in the further  enhancements |
| 2 | **Mentor’s Portal** | This feature allows  mentors to workout with  the leave/vacation  requests. | 6 | 1. Can approve a request.  2. Can decline a request.  3. Can edit the request.  4. Can verify the request. |
| 3 | **Warden’s Portal** | This feature allows  warden to authenticate  the requests | 6 | 1. Is able to view all requests  for outpass.  2. Can reconsider requests |
| 4 | **Hostel Supervisor’s**  **Portal** | This feature allows  supervisors to issue  outpasses to the students | 8 | 1. Can issue outpasses to  students.  2. Can deny the issue of an outpass.  3. Can send a request for  reconsideration to the  warden. |
| 5 | **Student Portal** | This feature will allow  students to raise a request for outing/extended  outing/leave. | 8 | 1. Can raise requests of  respective category.  2. Would receive a system  generated outpass |

# External Interface Requirement

User Interfaces

**Login Interface -** The login interface consists of the student username and password fields,

Student can login with the same VTop Credentials.

The login interface for the faculty and staff consists of the faculty/ staff employee id and

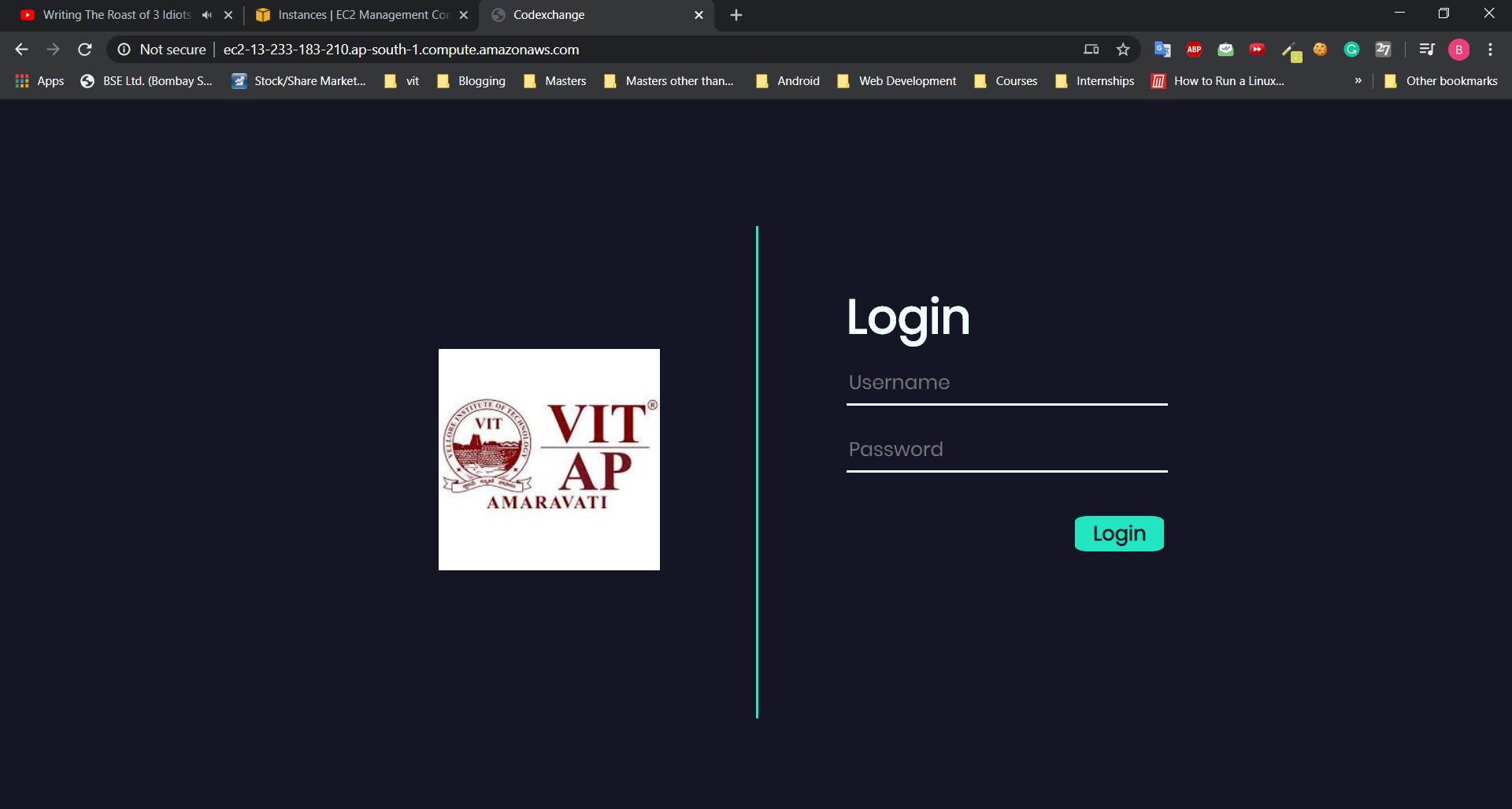
password. There credentials will also be the same as of VTop.

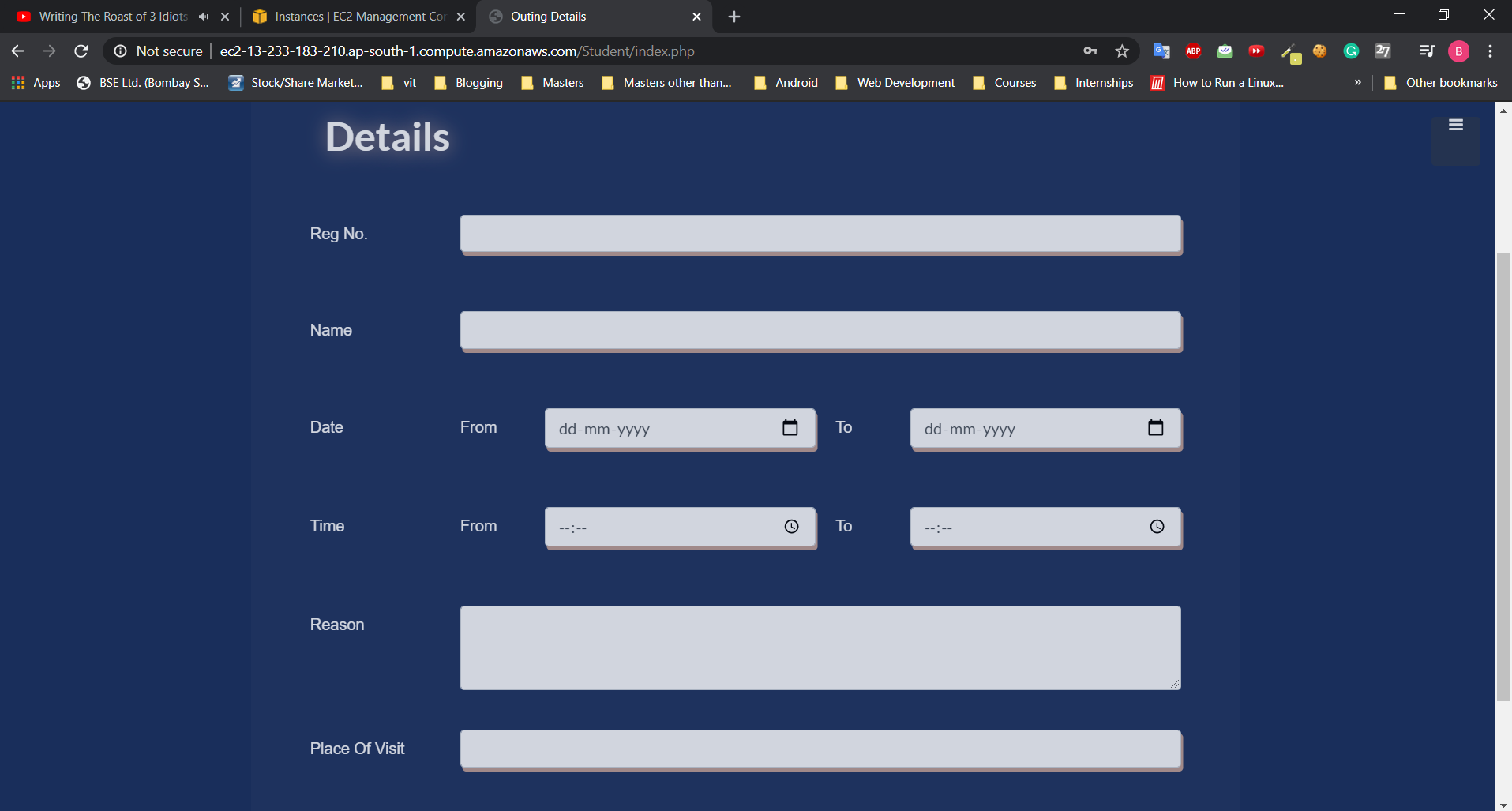
**Hardware Interfaces -** Hardware requirements include a laptop or a desktop or a smartphone with proper connectivity to access the system. Other than above mentioned, no hardware is required.

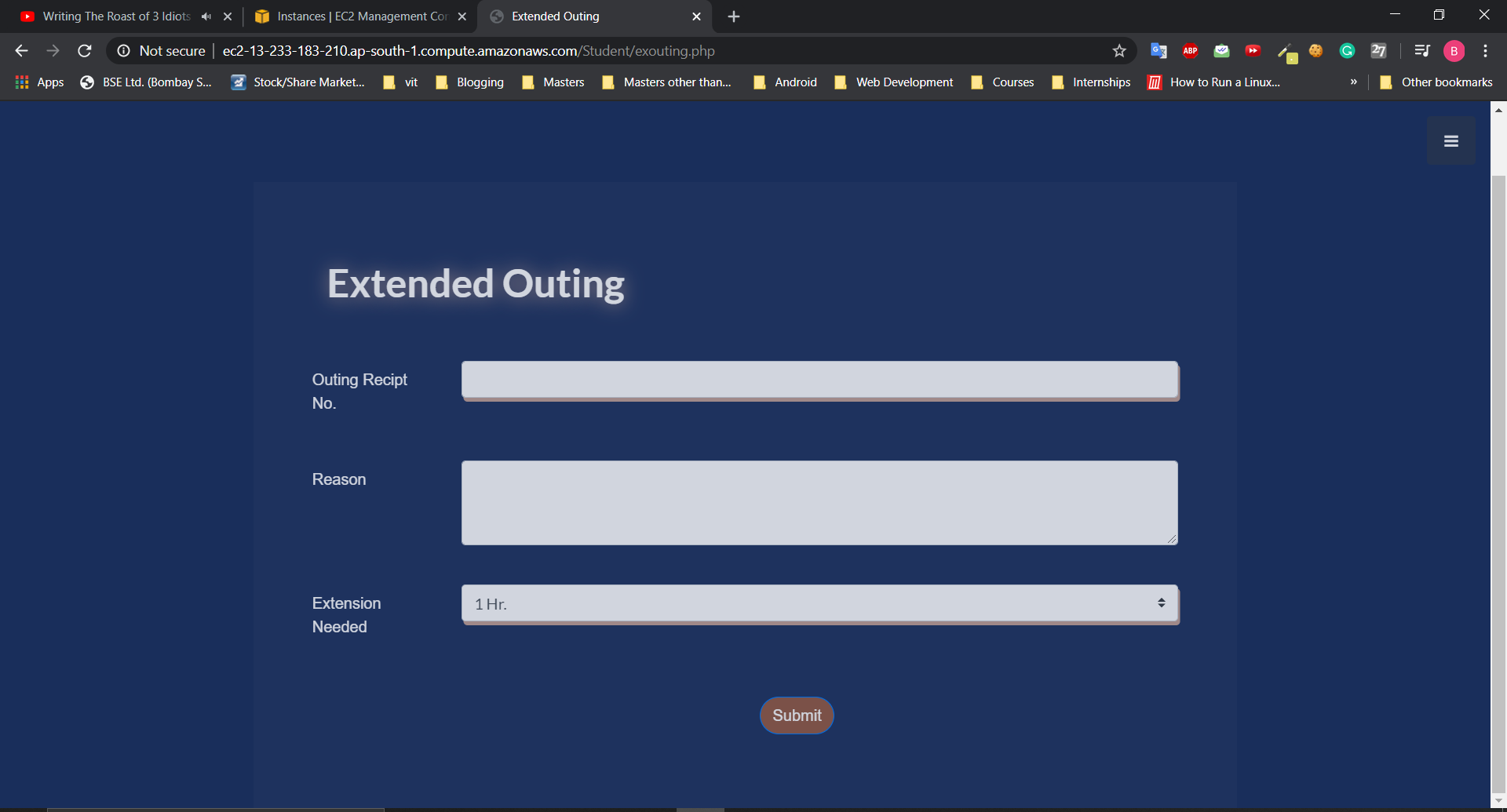
**Software Interfaces -** The software is based on application interface. The VITGo Application will interact with the VIT University Server with regard to user verification and information retrieval.

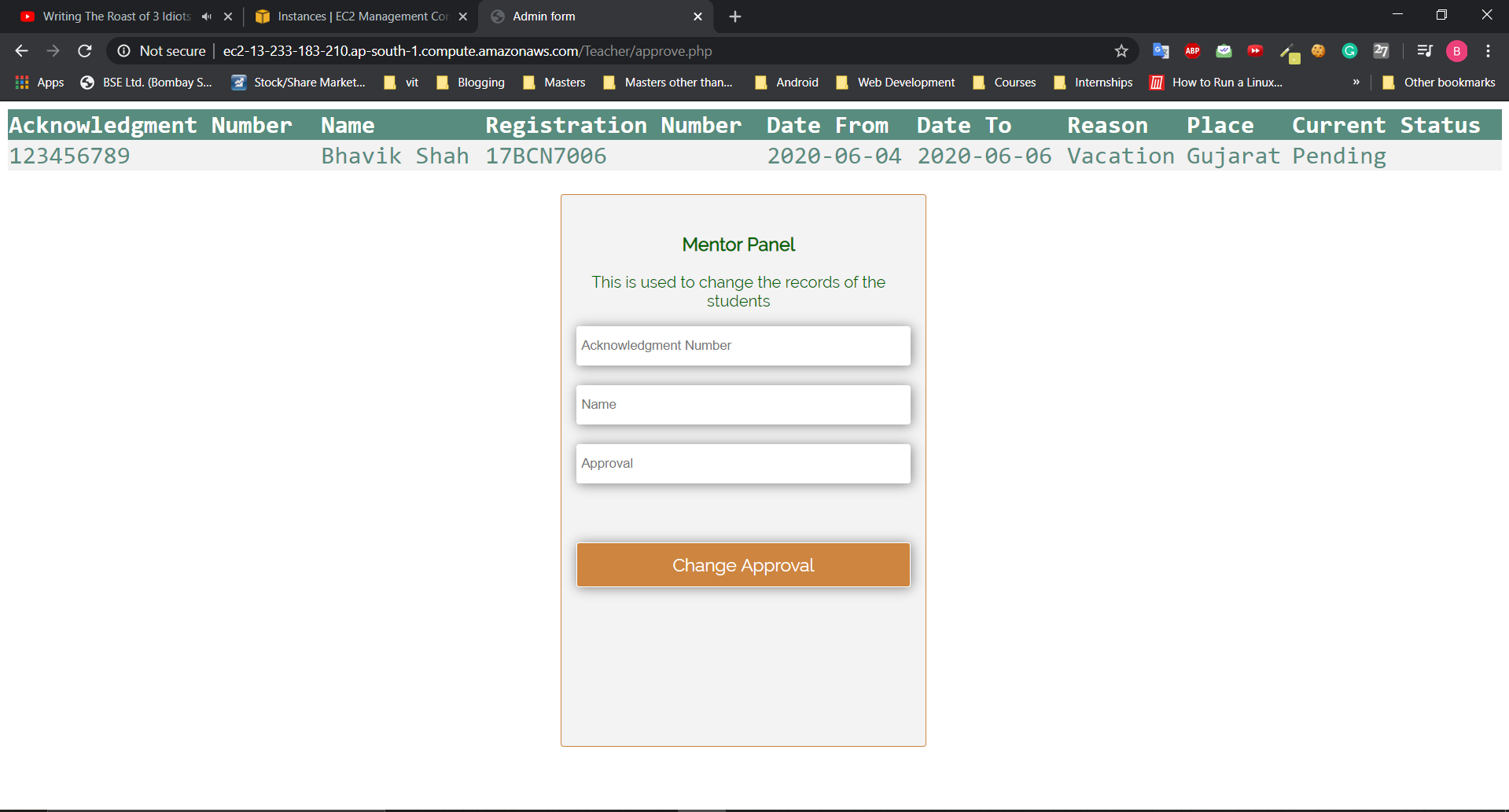
|  |  |
| --- | --- |
| Operating System | Ubuntu |
| Programming Language | HTML, PHP, CSS, JavaScript |
| IDE | Visual Studio Code |
| Database | InnoDB |
| Hosting Base | Amazon Web Services |

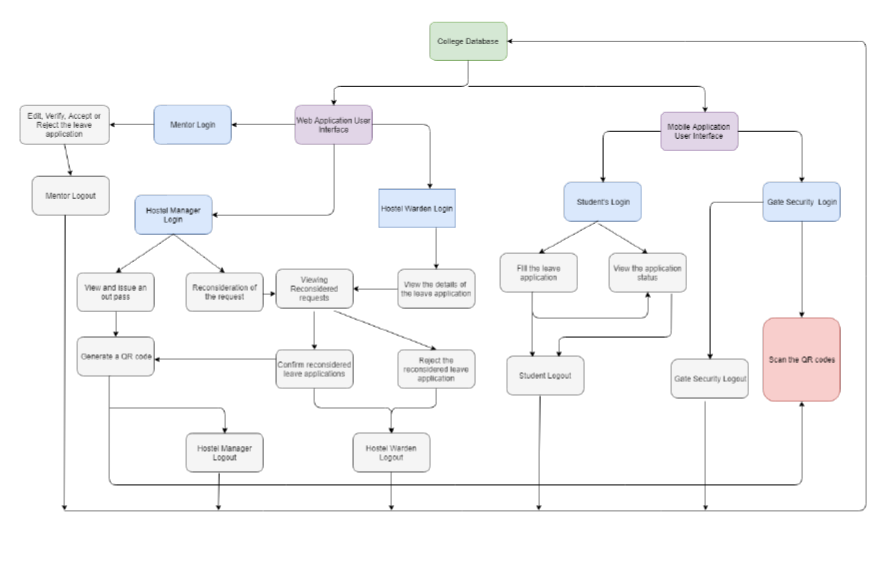
**Communication Interfaces -** This software would be functional on an ethernet connection or a wireless connection.

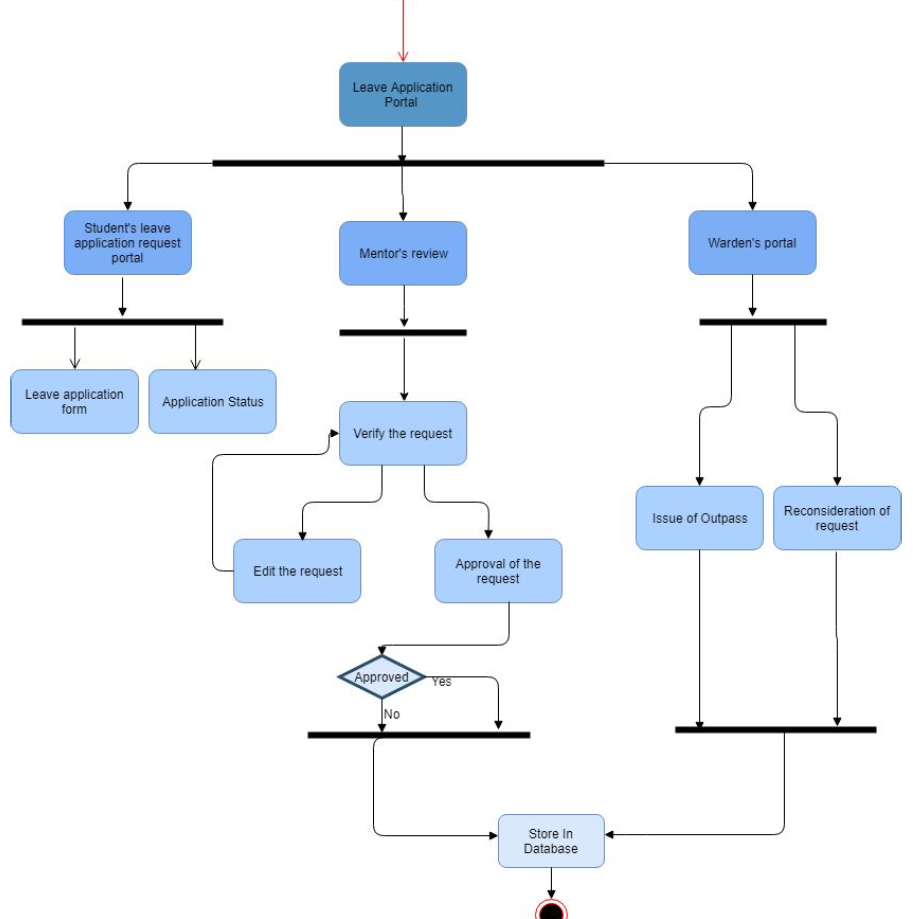


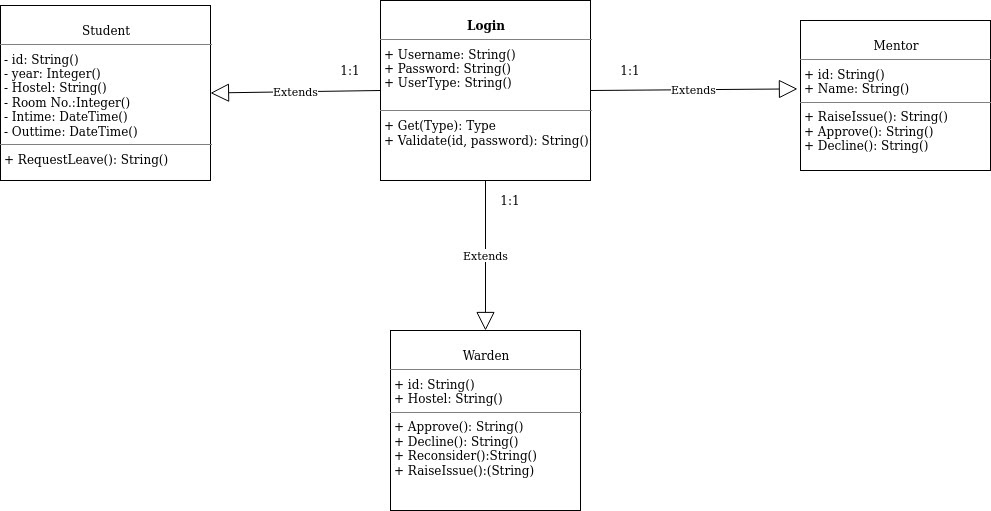


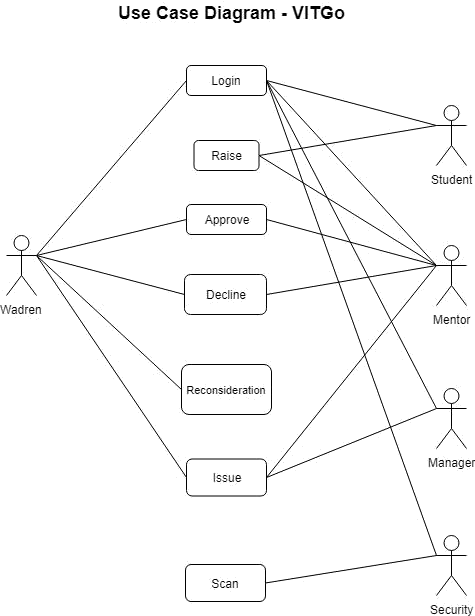


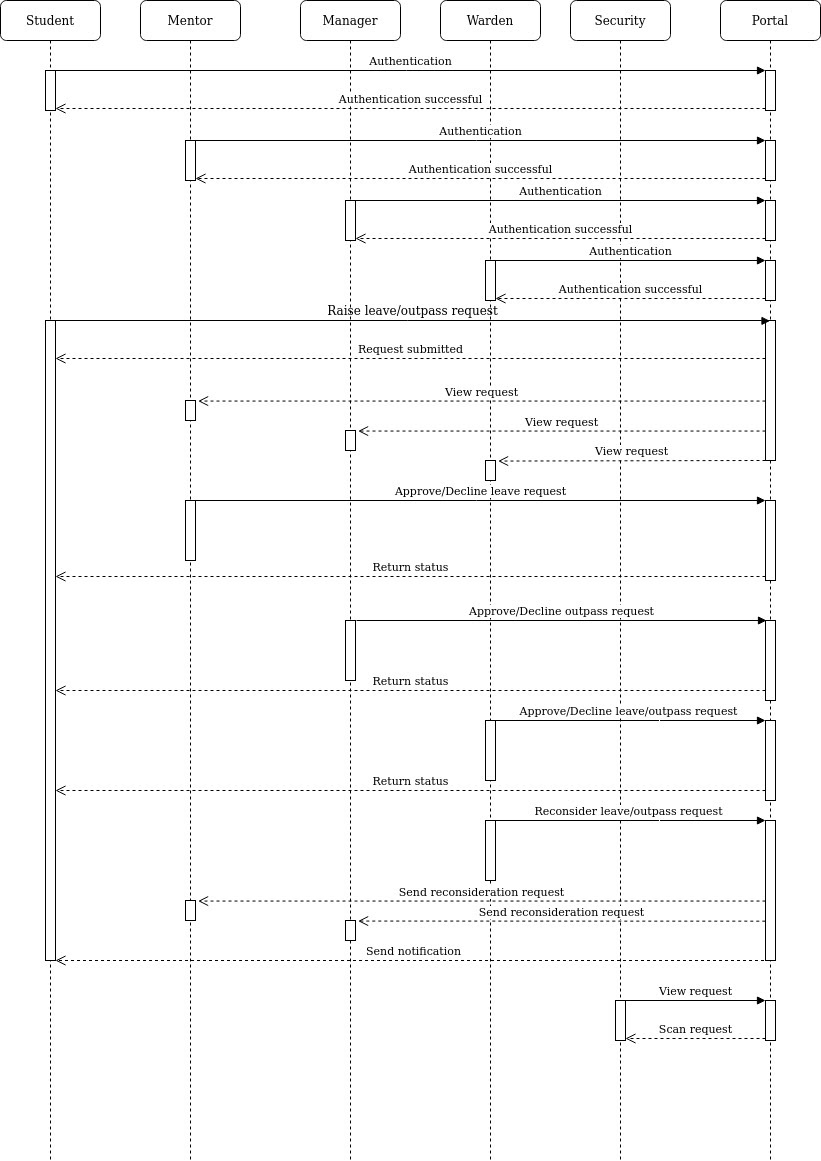












# Cost Calculation

# SOFTWARE COST ESTIMATION:

For any software project under development, it is indispensable to know how much it will cost to develop and how much development time will it take. The project scope must be established in advance and software metrics are used as a support from which evaluation is made. The project is broken into small PCs which are estimated individually. Several estimation procedures have been developed to monitor the project's progress, so developers and product managers can assess whether the project is progressing according to the procedure and take corrective actions, if necessary.

# STATIC, MULTIVARIABLE MODELS:

Static, multivariable models depend on several variables describing various aspects of the software development environment. In some models, several variables are needed to describe the software development process, and the selected equation combines these variables to give the estimate of time and cost.

WALSTON and FELIX developed the models at IBM to provide equations to give a relationship between lines of source code with effort and duration of development.

For our software project, the lines of code (LOC) sum up to 5223, which becomes 5.223 KLOC.

So, according to the WALSTON-FELIX model, we need to hire 24 engineers per month and require 7 and half months to develop our project.

# GITHUB LINK