**A**

**Major Project**

**On**

**Online Student Portal 2.0**

**Submitted in partial fulfillment for the requirement of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted to**



**RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA**

**2018-2022**

**Submitted By**

**AMAN JAIN** **ANKUSH JAIN**

**(0936CS181011)** **(0936CS181015)**

**DEEPESH PRAJAPATI** **NISHA SINGH**

**(0936CS181024)** **(0936CS181042)**

**Under the Guidance of**

**Prof. DILEEP SINGH RAJPUT**

**ASST. PROFESSOR (CSE)**



**Department of Computer Science & Engineering**

**DECLARATION**

We AMAN JAIN (0936CS1810011), ANKUSH JAIN (0936CS181015),

DEEPESH PRAJAPATI (0936CS181024), NISHA SINGH (0936CS181042) students of **B.TECH** **(Computer Science and Engineering)** IV Year, hereby declare that the synopsis entitled “**Online Student Portal 2.0**” which is being submitted to **Department of Computer Science and Engineering** in **Vikrant Institute Of Technology And Management, Gwalior** is our authentic work carried out for submitting as a synopsis during the VIII semester.

We declare that the work has not been submitted in partially or fully to any other University or Institute for the award of any Degree or Diploma.

AMAN JAIN

(0936CS181011)

ANKUSH JAIN

(0936CS181015)

DEEPESH PRAJAPATI

(0936CS181024)

NISHA SINGH

(0936CS181042)

**VIKRANT INSTITUTE OF TECHNOLOGY & MANAGEMENT**

**GWALIOR (MP)**

**Department Of Computer Science And Engineering**

**ACKNOWLEDGEMENT**

The success of any research study depends upon a number of factors among which the proper guidance from the experts in the industry and a faculty plays an important role.

We would like to express our heartfelt thanks to many people. This Project is an effort to contribute towards achieving the desired objectives. In doing so, we have optimized all available resources and made use of some external resources, the interplay of which, over a period of time, led to the attainment of the set goals.

We take here a great opportunity to express our sincere and deep sense of gratitude to **Prof. DILEEP SINGH RAJPUT** for giving us an opportunity to work on this project. The support & guidance from Sir, was of great help & it was extremely valuable.

We express our sincere thanks to all the people who, directly or indirectly, contributed in time, energy and knowledge to this effort.

**“Online Student Portal 2.0”**

**Submitted By: Submitted To:**

Aman Jain Prof. Dileep Singh Rajput

Ankush Jain Asst. Professor, CSE

Deepesh Prajapati

Nisha Singh

**Abstract :** This report specifies the various processes and techniques used in gathering requirements, designing, implementing for the project on OSP 2.0. This project aims to add some new functionalities to the previously implemented web application Online Student Portal. The requirements were gathered according to this system and this project is developed based on these requirements. This project was implemented in the form of a website using Django (Python).

**I. INTRODUCTION**

**Online Student Portal 2.0** is a vital part of any university’s running because students are what keep a University alive. The student’s management is one of the most important activities within a university as one cannot survive without students. A poor system can mean that the system is unreliable or unrelated because of mistakes or an overly slow response time. OSP 2.0 is the software which is helpful for registration of students and faculty as well as the colleges, the College is facilitated by the online system for registering students and faculty, add subjects and assignments in the department of students. Our System deals with the various activities related to the students. In the Software we can register as a user and user has of three types, college, student and faculty. Administrator has the power to add new user and delete a user. Through this online system we overcome many problems.

**II. EXPERIMENTAL PROCEDURES**

The Online Student Portal is a quite useful web application and can be implemented in a particular organization or institution. Here students are the main target users. Thus in this application all the features are executed successfully.

The portal is also includes notes section, To do section through which the student can manage records of their subject notes and day to day activities.

Books and Youtube sections are also available in this portal which can be beneficial for students as they can search information based on their requirements. If the student want to know meaning of any words, Dictionary section is also available in this project.

The whole process is divided into three modules College, Faculty and Student and unit testing was apply on each module. First unit testing was applied on college module and its functionalities each page was tested according to requirement. The testing was also applied on session to check whether the system automatically logout or not. Various operations like status active/ inactive, delete operation and update operation were applied on different tables. Filters were also tested whether they were working correctly or not. After all these methods, system testing was also applied on the whole portal.

**III. RESULTS**

All the modules were tested individually and the results shows that they were working properly. The working of this project was improved by using forms and filters. This project can be improved by adding more functionalities like library management and currently all the functionalities are working properly.

**CONCLUSION**

In this, we have described all the functionalities and modules of this project. Through this web based application the faculty can manage attendance and provide informative content to students. In future, many enhancement can be made on this project. It is concluded all the important point regarding this project.

**Refrences**

<http://www.w3school.com>

http://www.google.com

<https://stackoverflow.com>

https://www.tutorialspoint.com

**CERTIFICATE**

****

****

****

****

**TABLE OF CONTENTS**

**INTRODUCTION 12**

**OBJECTIVE 13**

**SCOPE 14**

*REASONS FOR RISING SCOPE OF ONLINE STUDENT PORTAL*

**SYSTEM REQUIRMENTS 15**

**OPERATING ENVIRONMENT 15**

**FRONTEND AND BACKEND TECHNOLOGIES 16-20**

**DESIGN APPROACH**

*ENTITY RELATION DIAGRAM* **21**

*DATA FLOW DIAGRAM* **22-24**

*USE-CASE DIAGRAM* **25**

**PROJECT MODULES 26**

*ADMINISTRATOR  
COLLEGE  
STUDENT*

**IMPLEMENTATION 27-28**

**TESTING 29-34**

**OUTPUT SCREENS 35-38**

**CONCLUSION 39**

**REFERENCES 40**

**INTRODUCTION**

**Online Student Portal 2.0** is a vital part of any university’s running because students are what keep a University alive. The student’s management is one of the most important activities within a university as one cannot survive without students. A poor system can mean that the system is unreliable or unrelated because of mistakes or an overly slow response time. OSP 2.0 is the software which is helpful for registration of students and faculty as well as the colleges, the College is facilitated by the online system for registering students and faculty, add subjects and assignments in the department of students. Our System deals with the various activities related to the students. In the Software we can register as a user and user has of three types, college, student and faculty. Administrator has the power to add new user and delete a user. Through this online system we overcome many problems.

1. Time and Money is saved.

2. Nothing is done manually.

3. Long lines in the department for these issues are resolved.

**OBJECTIVE**

The objective of this **OSP 2.0** is to allow the registration of students in particular course. It is intended to be complete specifications of what functionality the registration provides. It will also facilitate keeping all the records of students, such as their id, name, address etc. So all the information about a student will be available in a few seconds. Overall, it will make Student Online Registration System an easier job for the administrator and the student of any organization. The main purpose of this **OSP 2.0** document is to illustrate the requirements of the project Online Student Registration System and is intended to help any organization to maintain and manage its student’s personal data.

**SCOPE**

Without an Online Student Registration System, managing and maintaining the details of the student is a tedious job for any organization. **OSP** system will store all the details of the students including their information, educational qualifications, personal details and all the information related to their resume. System Automated process of registration is much better than the manual system as it has followings advantages: Time saving, increased efficiency, allows neat handling of data rather than error prone records, Decreases overhead and Accurate.

Thus all the features makes it best choice for the Institutions and organizations.

**SYSTEM REQUIREMENT**

**Software & Hardware Requirement**

Processor - Intel(R) Core™, Ryzen, or more….

System Type - 32bit or 64 bit Operating System

Operating System - Windows

RAM - 1024Mb or more…

Hard Drive - 500MB or more…

**Operating Environment**

Front-End - HTML5, CSS, JavaScript

Backend - Python and Django

Database - SQLite

**FRONTEND** – We have used HTML5, CSS3, and JavaScript for the development of website.

**HTML**

The **Hypertext Markup Language**, or **HTML** is the standard [markup language](https://en.wikipedia.org/wiki/Markup_language) for documents designed to be displayed in a [web browser](https://en.wikipedia.org/wiki/Web_browser). It can be assisted by technologies such as [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [scripting languages](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

[Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a [web page](https://en.wikipedia.org/wiki/Web_page) [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

[HTML elements](https://en.wikipedia.org/wiki/HTML_element) are the building blocks of HTML pages. With HTML constructs, [images](https://en.wikipedia.org/wiki/HTML_element#Images_and_objects) and other objects such as [interactive forms](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. HTML provides a means to create [structured documents](https://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](https://en.wikipedia.org/wiki/Semantics) for text such as headings, paragraphs, lists, [links](https://en.wikipedia.org/wiki/Hyperlink), quotes and other items. HTML elements are delineated by *tags*, written using [angle brackets](https://en.wikipedia.org/wiki/Bracket#Angle_brackets). Tags such as <**img** /> and <**input** /> directly introduce content into the page. Other tags such as <**p**> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can embed programs written in a [scripting language](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript), which affects the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content. The [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

**CSS – Cascading Style Sheet**

**Cascading Style Sheets** (**CSS**) is a [style sheet language](https://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [presentation](https://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](https://en.wikipedia.org/wiki/Markup_language) such as [HTML](https://en.wikipedia.org/wiki/HTML). CSS is a cornerstone technology of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), alongside HTML and [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

CSS is designed to enable the separation of presentation and content, including [layout](https://en.wikipedia.org/wiki/Page_layout), [colors](https://en.wikipedia.org/wiki/Color), and [fonts](https://en.wikipedia.org/wiki/Typeface). This separation can improve content [accessibility](https://en.wikipedia.org/wiki/Accessibility); provide more flexibility and control in the specification of presentation characteristics; enable multiple [web pages](https://en.wikipedia.org/wiki/Web_page) to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be [cached](https://en.wikipedia.org/wiki/Cache_(computing)) to improve the page load speed between the pages that share the file and its formatting.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or [screen reader](https://en.wikipedia.org/wiki/Screen_reader)), and on [Braille-based](https://en.wikipedia.org/wiki/Braille_display) tactile devices. CSS also has rules for alternate formatting if the content is accessed on a [mobile device](https://en.wikipedia.org/wiki/Mobile_device).

The name *cascading* comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

**JAVASCRIPT**

**JavaScript** is a lightweight, cross-platform and interpreted scripting language. It is well-known for the development of web-pages, many non-browser environments also use it. JavaScript can be used for **Client-side** developments as well as **Server-side** developments.

**Editor: Visual Studio Code**

Visual Studio Code (VS Code) is a source code editorial manager created by Microsoft that can be run on Windows, macOS, and Linux. It is free, open-source, and offers help for investigating just as implicit Git variant control, linguistic structure features, scraps, etc. The UI of the VS Code is exceptionally adjustable, as clients can change to various subjects, console alternate ways, and inclinations.

**BACKEND** – We decided to make the backend of our project using Python with Django.

**Python (Programming Language)**

**Python** is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [high-level](https://en.wikipedia.org/wiki/High-level_programming_language) [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). Its design philosophy emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability) with its use of [significant indentation](https://en.wikipedia.org/wiki/Off-side_rule). Its [language constructs](https://en.wikipedia.org/wiki/Language_construct) as well as its [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) approach aim to help [programmers](https://en.wikipedia.org/wiki/Programmers) write clear, logical code for small and large-scale projects.

Python is [dynamically-typed](https://en.wikipedia.org/wiki/Type_system#DYNAMIC) and [garbage-collected](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science)). It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [structured](https://en.wikipedia.org/wiki/Structured_programming) (particularly, [procedural](https://en.wikipedia.org/wiki/Procedural_programming)), object-oriented and [functional programming](https://en.wikipedia.org/wiki/Functional_programming). It is often described as a "batteries included" language due to its comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library).

[Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum) began working on Python in the late 1980s, as a successor to the [ABC programming language](https://en.wikipedia.org/wiki/ABC_(programming_language)), and first released it in 1991 as Python 0.9.0.Python 2.0 was released in 2000 and introduced new features, such as [list comprehensions](https://en.wikipedia.org/wiki/List_comprehension) and a [cycle-detecting](https://en.wikipedia.org/wiki/Cycle_detection) garbage collection system (in addition to [reference counting](https://en.wikipedia.org/wiki/Reference_counting)). Python 3.0 was released in 2008 and was a major revision of the language that is not completely [backward-compatible](https://en.wikipedia.org/wiki/Backward_compatibility). Python 2 was discontinued with version 2.7.18 in 2020.

**Django**

Django is an advanced Web framework written in Python that makes use of the model view controller (MVC) architectural pattern. Django was created in a fast-moving newsroom environment, and its key objective is to ease the development of complicated, database-driven websites. This Web framework was initially developed for The World Company for managing some of their news-oriented sites.

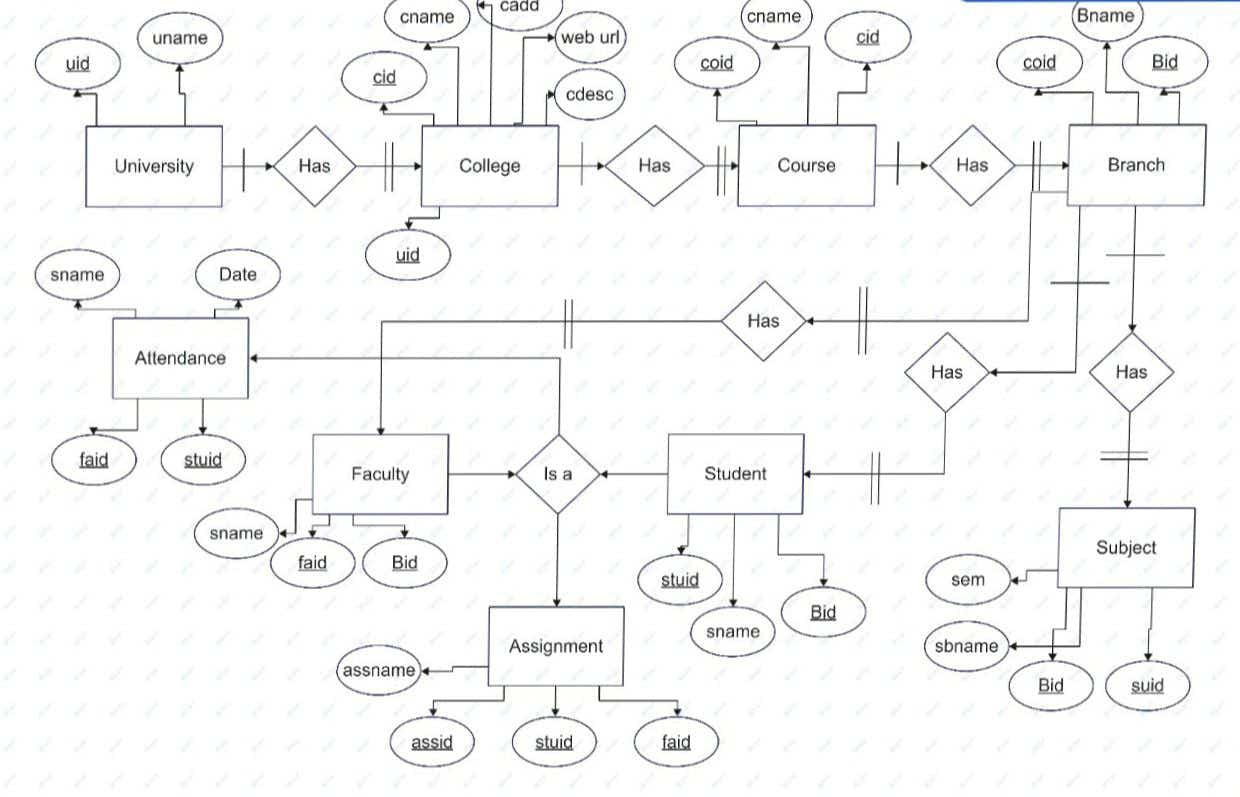
Django is available as an open-source Web framework, and it uses Python extensively to create files, settings and data models. It is designed to address two main challenges: the rigorous requirements of highly experienced Web developers and the intense deadlines of a newsroom. Django concentrates more on automating wherever possible and sticking to the "don't repeat yourself" principle.

Django emphasizes the following:

* Plug ability and reusability of components
* Quick development
* The principle of non-repetition

**DESIGN APPORACH**

**E- R DIAGRAM**

****

**DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a structured analysis and design tool that can be used for flowcharting. A DFD is a network that describes the flow of data and the processes that change or transform the data throughout a system. This network is constructed by using a set of symbols that do not imply any physical implementation. It has the purpose of clarifying system requirements and identifying major transformations. So it is the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. DFD can be considered to an abstraction of the logic of an information-oriented or a process-oriented system flow-chart. For these reasons DFD’s are often referred to as logical data flow diagrams.

EXTERNAL ENTITY

An external entity is a source or destination of a data flow. Only those entities which originate or receive data are represented on a data flow diagram. The symbol used is a rectangular box.

PROCESS

A process shows a transformation or manipulation of data flow within the system. The symbol used is an oval shape.

DATA FLOW

The data flow shows the flow of information from a source to its destination. Data flow is represented by a line, with arrowheads showing the direction of flow. Information always flows to or from a process and may be written, verbal or electronic. Each data flow may be referred by the processes or data stores at its head and tail, or by a description of its contents.

DATA STORE

A data store is a holding place for information within the system. It is represented by an open ended narrow rectangle. Data stores may be long-term files such as sales ledgers, or may be short-term accumulations; for example batches of documents that are waiting to be processed. Each data store should be given a reference followed by an arbitrary number.

**DATA FLOW DIAGRAM (DFD)**

**College/ Faculty / Student Login**

**Existing User**

**No**

**Create New Account**

**College/Student/Faculty**

**Yes**

**User Type**

**Admin**n

**Faculty**

**Student**

**College**

**Download/ Submit Assignment**

**Mark Attendance Of Students**

**Faculty Register**

**Register University**

**Student Register**

**Show Attendance**

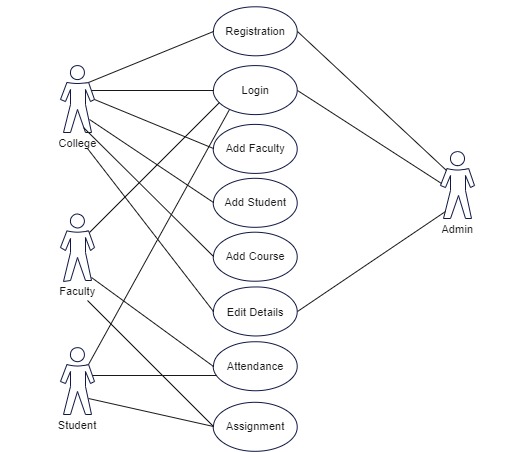
**Give / Check Assignment Of Students**

**College Status Update / Delete**

**Manage Faculty And Students**

**Student Section**

**USE CASE DIAGRAM**

****

**PROJECT MODULE**

**IDENTIFICATION OF STAKEHOLDERS**

Before implementation of this project, we need to identify the various stakeholders of the project which are as follows :-

* **ADMINISTRATOR**

The administrator is the super user of this web application. Only admin have access into this admin page. Admin may be the owner of the website. The administrator has all the information about all the College, Faculty and Student.

* **COLLEGE**

College are the key stakeholders of this web application. They can login their account by providing required credentials. They are responsible to manage and edit their own details. They can also manage the Faculty and Students.

* **FACULTY**

Faculty are another key stakeholders of this web application. They can login their account by providing required credentials. They are responsible to manage and edit their own details. They can provide informative content to their respective students. They can also manage the details of students.

* **STUDENT**

Student are the end user of this web application. They can login their account by providing required credentials. They can take assignments of their particular colleges from their home.

**IMPLEMENTATION**

There are mainly three modules to implement in this project.

**Administrator** – The administrator should have the following features.

1. Register universities
2. Authorize the registered colleges
3. Delete the account of college.

**College** – The College have different table to implement for the effective working of this project. These tables are

1. College
2. Students
3. Courses
4. Branches
5. Subject
6. Assignment

The college should have the following features.

* Register themselves by providing required information
* Logging in their account by providing their own user id and password
* Register their students by providing required information
* Add and update their own details
* Add and update their courses, branches and subjects
* Provide assignment to their registered students.

**Faculty** -They should have following features :

* Logging in their account by providing their own user ID and Password.
* Add and edit their details like number, address etc.
* Faculty can manage student based on their subject.
* Faculty can manage attendance of students.
* Faculty can provide informative content to their students.
* Faculty can check information submitted the students.

**Student** – The students should have the following features

* Logging in their account by providing their own user id and password
* Add and update their own details
* Download their assignments from their respective colleges
* Submit their to their respective colleges.

**TESTING**

Testing is a set of activities that can be planned in advance and conducted systematically. Testing requires that the developer discard preconceived notations of the correctness of the software just developed and overcome a conflict of interest that occurs when errors are encountered. Testing also provides the main objective of our project and understands the risk of implementation. Testing is a process of technical investigation, performed on behalf of stakeholder that is intended to reveal quantity related information about the product with respect to the context in which it in intended to operate. Testing is the process of executing a program or an application with intent of finding an error or bugs. Testing can be stated as the process of validating and verifying that a software program/application. Ideally, testing will exercise the system in all possible ways.

**Test Plan:-**

Test plan will describe about the scope and activities of our modules in the project. We must plan the test plans in the starting of our quiz project. It will provide a unique identifier for our document. Testing should begin in “small” and proceeds in the “large”. Exhaustive testing is not possible. Provides an overview of our test plan. Specify the goals/ objective/ constraints. Our project has six modules. So, we will do testing from the starting phase itself. The main aim of test plan is to produce a correct code with all users requirements satisfied.

- Covers all types and phases of testing & guides the entire testing process.

- Who, why, when, what

- Developed as requirements, functional specification, and high-level design are developed & should be done before implementation starts.

A test plan includes test objectives, schedule and logistics, test strategies, test cases procedure, expected result, procedures for handling problems.

**1. UNIT TESTING:-**

Unit testing is focused on verifying small portions of functionality. In practice, it’s usually looks a lot like debugging & more structured approaches exist in our project. For example, an individual unit test case might focus on verifying that the correct data has been saved to the database when the create, modify, delete button on particular page is clicked. An important sunset of unit testing that is often overlooked is range checking. That is, making sure all the fields that collect information from the user, can gracefully handle any value that is entered. Most people think of range checking as making sure that a numeric field only accepts numbers. In addition to traditional range checking make sure you also check for less common, but just as problematic expectations. For example, what happens when a user enters his or her first name and the first name contains an apostrophe, such as O’Brien? Different combinations of database and database drivers handle the apostrophe differently, sometimes with unexpected results. Proper unit testing will help rid your web application of obvious errors that your users should never have to encounter.

**User-Interface Testing:-**

1. a) All fonts should be same as per templates.
2. b) All the error messages should be correct without any spelling or grammatical errors and the error message should match with the field label.
3. c) Tool tip text should be there for every field.
4. d) Enough space should be provided between field labels, columns, rows, and error messages. e) All the buttons should be in a standard format and size.
5. f) Disabled fields should be grayed out.
6. g) Check for broken links and images, if any.
7. h) Confirmation message should be displayed for any kind of update and delete operation.
8. i) Check the end user can fill entire form.
9. j) Check the tab sequence and it should work properly.
10. k) Scroll bar should appear only if required.
11. l) If there is an error message on submit, the information to be filled by the user should be prompted.
12. m) Title should display on each web page.
13. n) Web page content should be correct without any spelling or grammatical errors.

**Functional Testing:-**

1. a) Test all the mandatory fields should be validated against system dates.
2. b) Test the asterisk sign should display for all the mandatory fields.
3. c) Test the system should not display the error message for optional fields.
4. d) Test the numeric fields should not accept the alphabets and proper error message should display.
5. e) Test for negative numbers if allowed for numeric fields.
6. f) Test the max length of remarks/comments field to ensure the data is not truncated.
7. g) Test that a confirmation message should display for update and delete operations.
8. h) Test if any functionality fails the user gets redirected to the custom error page yet to be developed.
9. i) Test all the data inside combo/list box is arranged/sorted in chronological order.

**Database Testing:-**

1. a) Verify whether the column allows a null or not.
2. b) Verify the Status of Stored Procedure(valid/invalid).
3. c) Test the parameters if they are required or not.
4. d) Test when the report output is zero for a particular item, the zero records should not be affected.
5. e) Test the table row with duplicate/identical sample input data values.

f) Verify the data gets properly saved into table(s) of the database after the each page submission.

1. g) Verify the data if the DML (Update, delete and insert) operations are performed in the output report.
2. h) Check the length of every field: The field length in the back-end table and front-end form must be same.
3. i) Verify the report output with data entered in conjunction with previous data values.
4. j) Verify the roles in encrypted data in the database.
5. k) Verify the database size. Also observe the response time of each submission query executed.
6. l) Verify the data displayed on the front end and make sure it is same in the back end.
7. m) Verify the data validity by inserting the invalid data in the database.
8. n) Verify the execution impact of triggers in other table/views.

**2. INTEGRATION TESTING:-**

Integration testing (sometimes called Integration and Testing, abbreviated “I & T”) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing, integration testing takes as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules like login, employee, designation, increment, salary that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing. Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. Scope of testing is summarizes the specific functional, performance and internal design characteristics that are to be tested. Integration testing is a type of software that seeks to verify the interface between components. The components are integrated in an iterative way. Using integration testing will allow the interface to find the issues more quickly and fixed in our project. It works to expose defects in the interfaces and interaction between integrated components.

**3. SECURITY TESTING:-** - Authorization- Access control, user permission are an example of authorization.

- Authentication – Password, Biometric (unique id) are an example of Authentication. - Availability- Software is available to authorized persons when they need it outside the maintenance window.

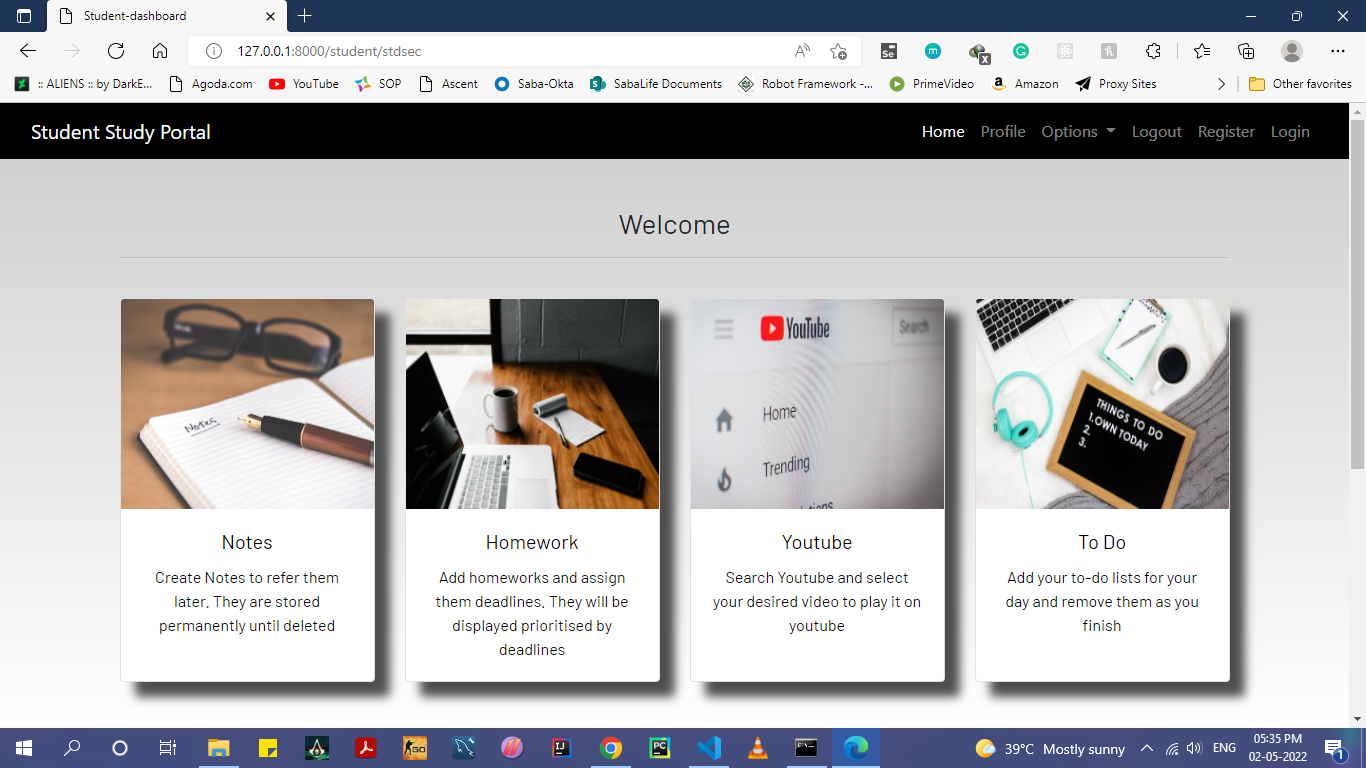
- Confidentiality – Protection of information and resources from the users other than the authorize and authenticated.

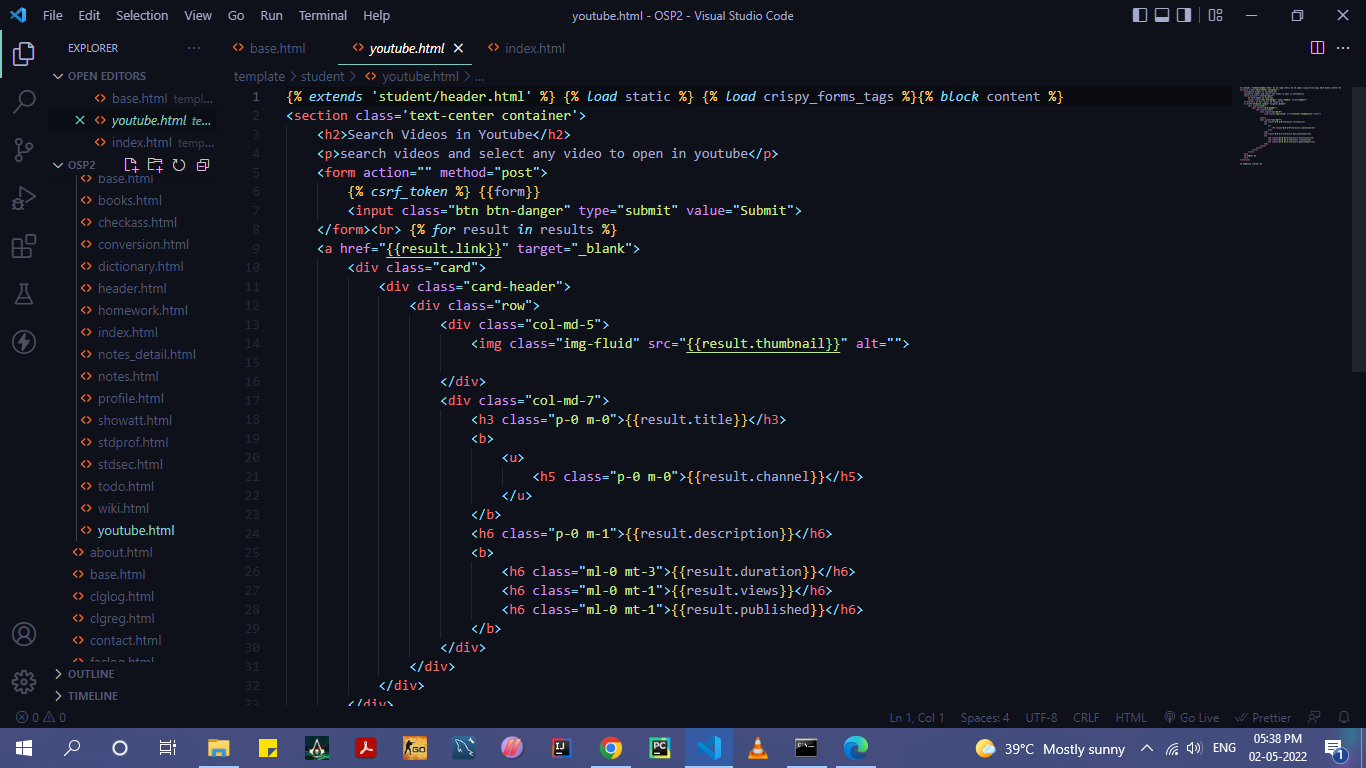
- Integrity – Ensures the information received is not altered during the transit.

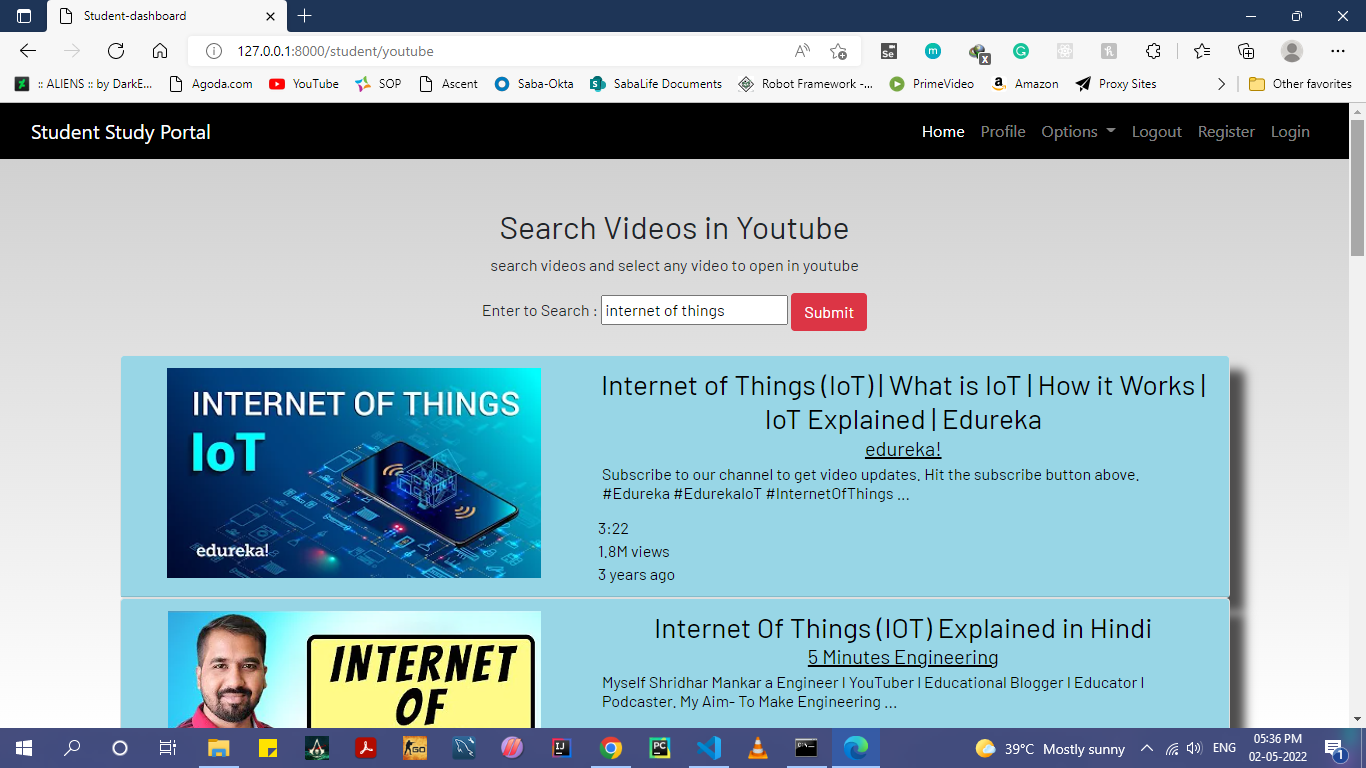
Security testing involves tests to determine how the software/application protect data, restricts unauthorized access and maintains functionally as intended. Security testing is more crucial for client server and web applications. The primary reason for testing the security to identify potential vulnerabilities and subsequently repair them. Below are the five basic security concepts that need to be covered by security testing.

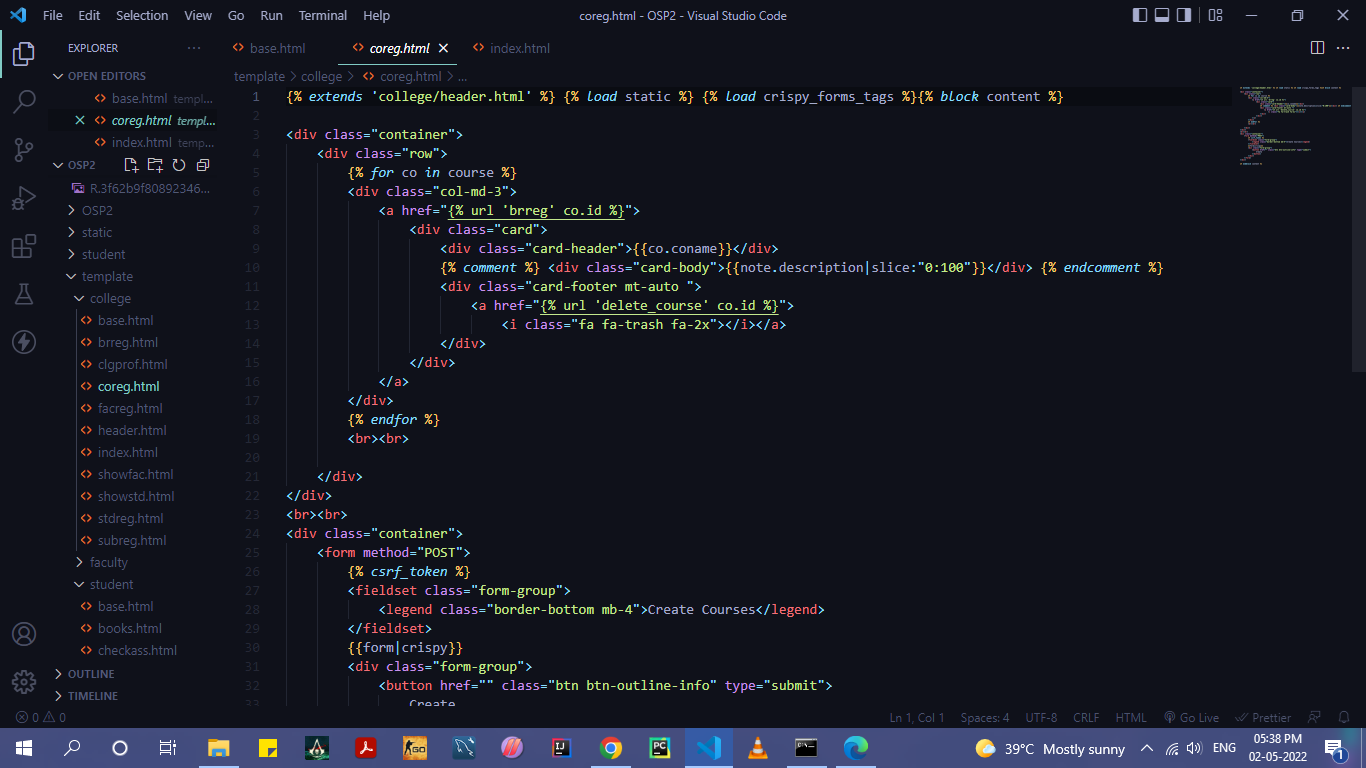
E.g.:- our software availability is 24 X 7 outside the maintenance window.

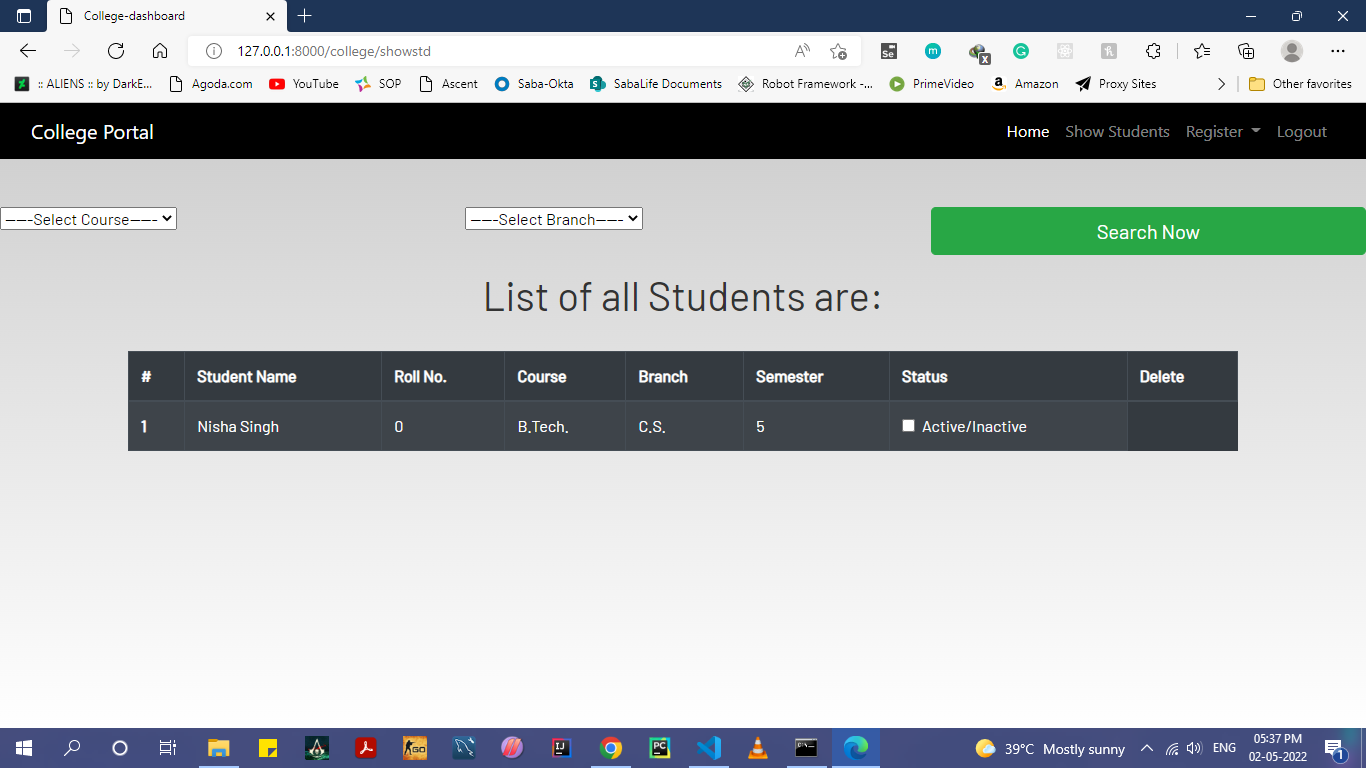
**OUTPUT**

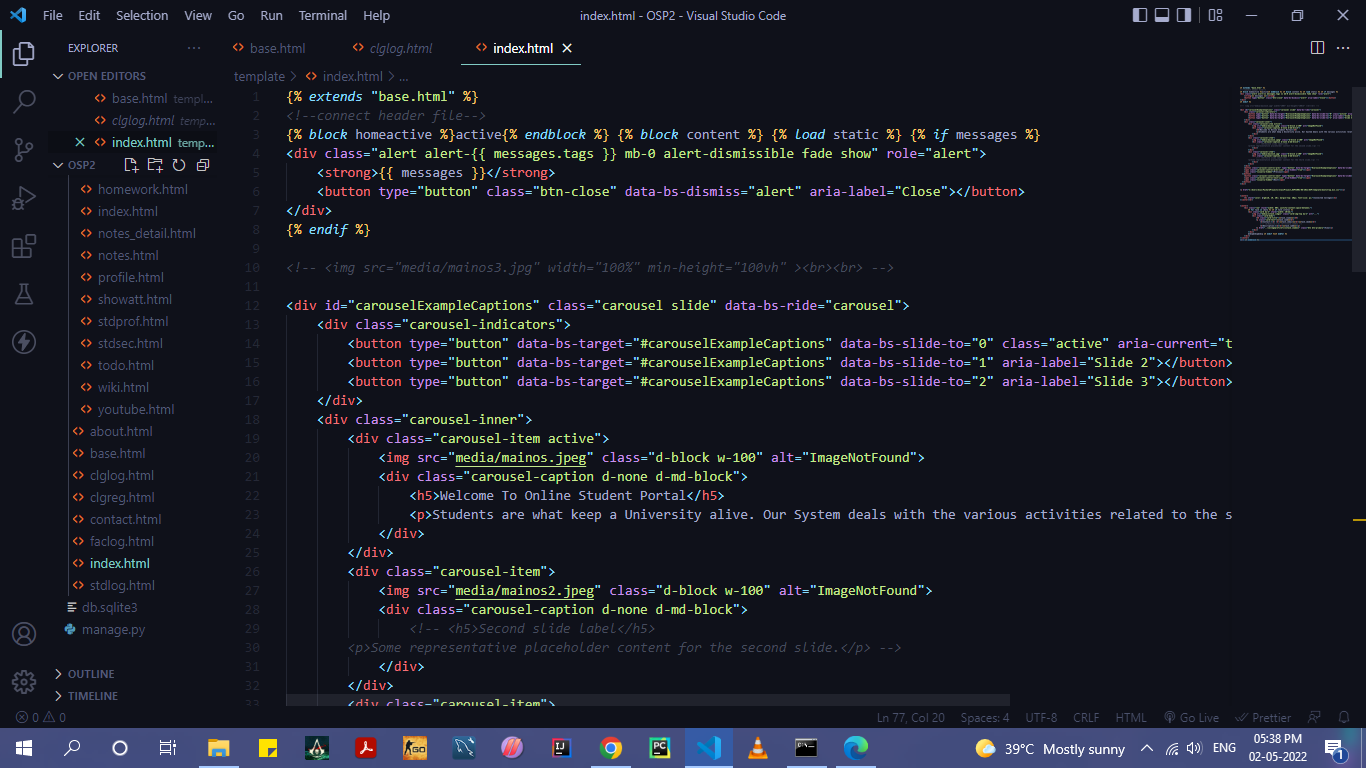
****

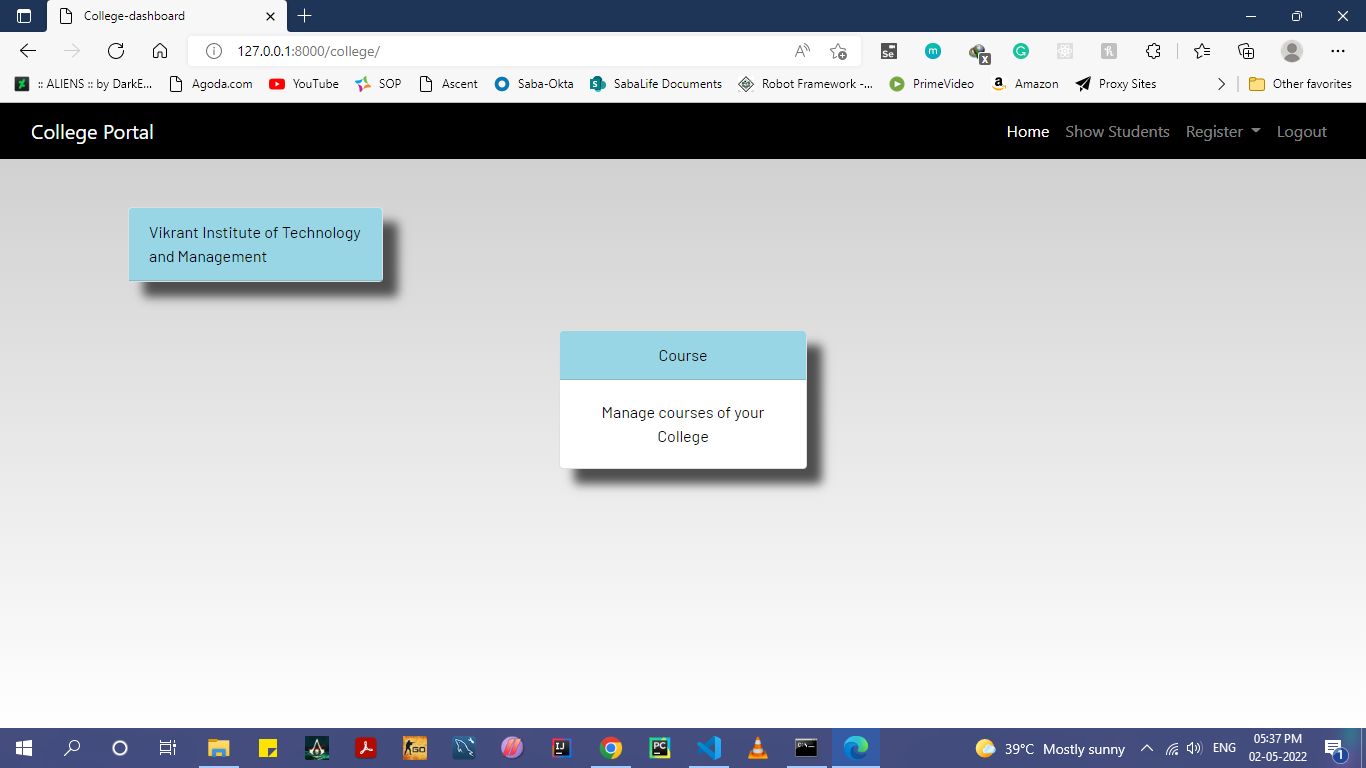
****

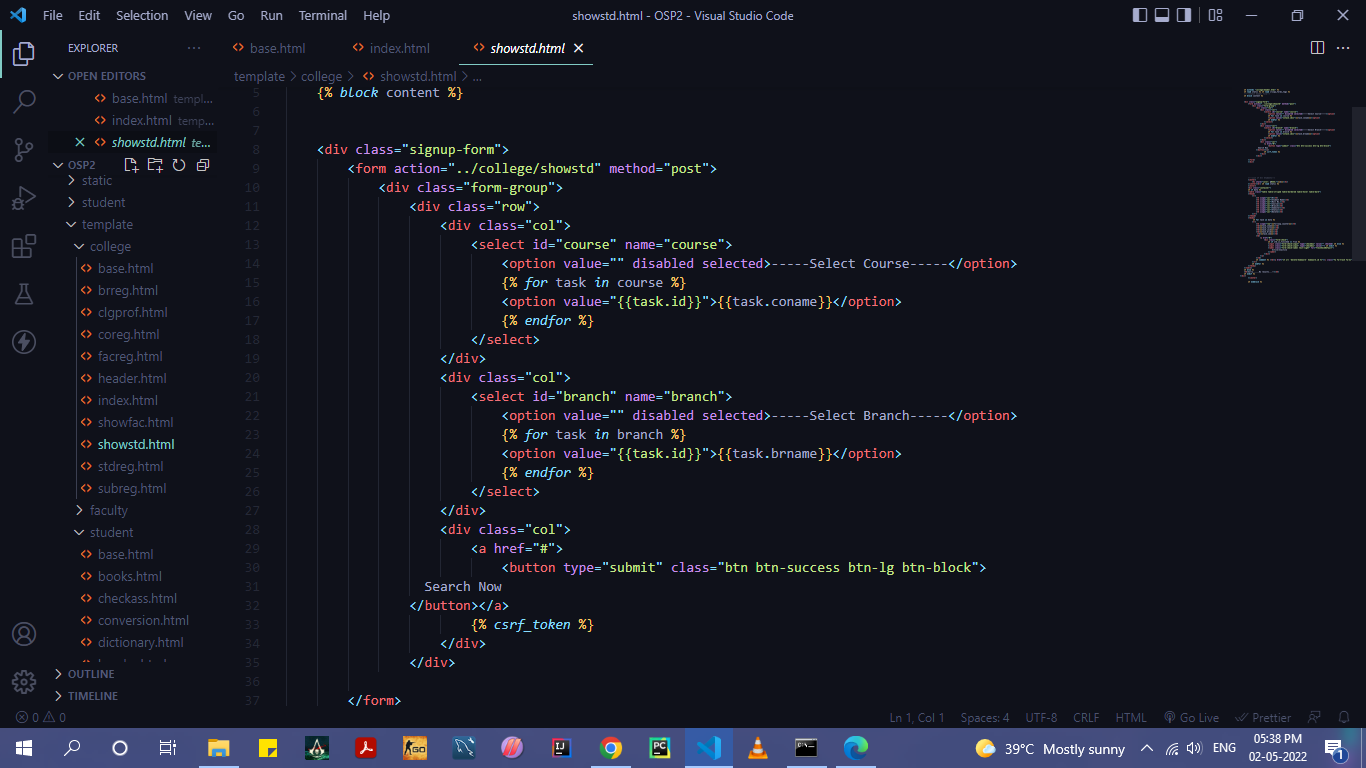












**CONCLUSION**

In this, we have described all the functionalities and modules of this project. Through this web based application the faculty can manage attendance and provide informative content to students. In future, many enhancement can be made on this project. It is concluded all the important point regarding this project.

**REFERENCES**

* <http://www.w3school.com>
* http://www.google.com
* <https://stackoverflow.com>
* https://www.tutorialspoint.com