**ABSTRACT**

Enterprise useful resource planning (ERP) refers to a kind of software program that groups use to control every day enterprise sports which includes accounting, procurement, venture control, danger control and compliance, and deliver chain operations. This venture is primarily based totally at the idea of numerous university control structures and Enterprise Resource Planning structures utilized by universities. It manages the university statistics, scholar statistics, placement statistics, and specific occasions happening withinside the university. It additionally continues a file of all of the statistics concerning college students who've gotten positioned or are withinside the technique of placement. It has a word board which includes statistics approximately numerous cultural or technical or any sports activities which can be speculated to be held soon. College control structures also can generate scholar overall performance reports &amp; simplify the hassles of school members. It may even be used for on-line admissions, digital coaching, promoting courses, monitoring scholar progress, assessments and quizzes, dealing with attendance, on-line assignments, and lots more. Its most important motive is to efficiently and efficaciously create and keep environments inside instructional establishments that promote, support, and preserve powerful coaching and learning.

**CHAPTER-1**

**Introduction**

* 1. **Introduction**

Enterprise resource planning (ERP) refers to a sort of software program that agencies use to manipulate everyday enterprise activities which includes accounting, procurement, undertaking control, risk control and compliance, and deliver chain operations. A whole ERP suite additionally consists of organisation overall performance control, software program that allows plan, budget, predict, and file on an organization`s monetary results.

ERP structures tie collectively a large number of enterprise strategies and allow the glide of statistics among them. By gathering an organization`s shared transactional statistics from a couple of sources, ERP structures get rid of statistics duplication and offer statistics integrity with a unmarried supply of truth.

Today, ERP structures are essential for coping with hundreds of corporations of all sizes and in all industries. To those companies, ERP is as integral because the energy that maintains the lighting on.

* 1. **Existing System**
     1. Metafresh - Metafresh is a laboriously actively maintained fork of Adempiere and may be used and allotted freely. It does now no longer require a contributor license settlement from companions or contributors. Features of metafresh:
        + Digital Sovereignty
        + Ready for Immediate Use
        + Unique Business Model
        + Freedom & Scalability
        + Interconnectedness
        + Digitalization
        + Multitenancy
     2. Apache OFBiz **-** Apache OFBiz is a suite of business operations flexible enough to be used across any assiduity. A common armature allows inventors to fluently extend or enhance it to produce custom features.

Apache OFBiz has a full set useful business features including

● Product & roster operation

● Promotion & Pricing operation

● Supply Chain Fulfillment

● Contracts, Payments & Billing

* + 1. iDempiere - It is the perfect tool for folks trying to find an Enterprise Resource designing (ERP) package that's open supply.

Main options of iDempiere are:

● Purchase and liabilities Management

● Supports most OS and Platforms

● Contract Management

● Production designing

● Unified Dashboard

● Unified management Centre

● Financial and Accounting Management

● Supports most OS and Platforms

● Customer Relationship Management

● Sales and selling Management

* + 1. DoliDroid - DoliDroid is a front-end Android client of the Dolibarr ERP & CRM web software. This application is not a standalone program. This is the user interface to use on your Dolibarr ERP & CRM software hosted online. The advantages of DoliDroid are:

● DoliDroid uses embedded image resources to reduce bandwidth.

● DoliDroid uses internal cache for unchanged pages.

● Connection settings are saved. No need to import them every time you use DoliDroid.

● Better integration with your phone or other apps (click open PDF PDF reader, click email or phone, launch your email client or launch Android dialer.

* 1. **Problems in Existing Definition**
     1. The application logo being used is not attractive or eye catching
     2. The title of the application doesn’t let the user know what the function of the application is.
     3. The features mentioned include some bugs and data loading issues, many of which have no results on startup.
     4. The overall look and feel of the application is that of a low quality product as the user interface does not look professional or user friendly.
     5. The Time Table feature should include the latest and updated information for the ease of students as well teachers. It can even include the option of notifying students in case of cancellation of any lecture.
     6. Course registration function should include a brief description of the courses which will make it easier for students to opt for them.
     7. Having a College Events Calendar will help the students plan out their activities and schedules while also help them make a note of important dates.
  2. **Problem Definition**

To improve the overall look and feel of the existing college management system while including multiple ease of life features like E-Id card, course resources, interactive campus map, placement updates and information, e-library and much more.

* 1. **Feasibility Study**

There are n number of ERP systems as well as multiple college management softwares available in the market. Many of these systems require a large amount of resources to build and deploy but it is still possible to achieve this outcome using miniscule amount of storage and computing hence it is feasible to bring this project to fruition.

* 1. **Motivation**

The motivation to pursue this project comes from the existing ERP system used by Manav Rachna International Institute of Research and Studies as well as from various “Super Apps” in use right now for e.g. WeChat, Paytm, Grab etc. Super Apps refers to those applications that provide facilities of multiple domains on a single platform.

* 1. **Project Overview**

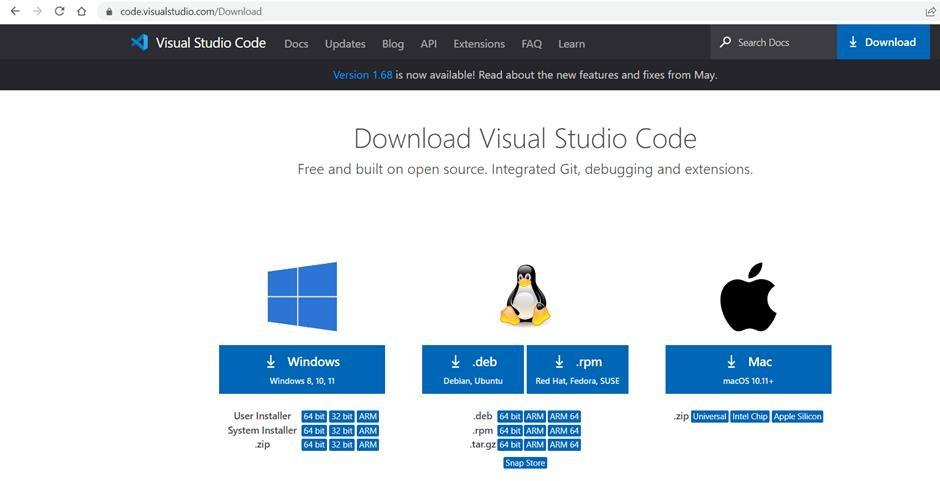
This project will cover existing features of college management systems and on top of that some unique ones that have not yet been deployed in the current systems for eg. Geolocation, Placement support, Course resources, E-id card etc.

The techstack used in order to build this project includes IDEs like PyCharm Community Edition, Visual Studio Code, front end languages like HTML and CSS, Python3 and Flask Framework along with MySQL for database management.

* 1. **Hardware Specification**
     + - A system with Windows/Linux operating System.
       - Processor with more 1.7gHz speed.
       - Minimum 4gb of Memory.
       - Minimum 5gb of empty Storage.
       - An Integrated Graphic card.
  2. **Software Specification**
     1. **Visual Studio Code** - Visual Studio Code, additionally usually known as VS Code, is a source-code editor made via way of means of Microsoft for Windows, Linux and macOS. It consists of functions like aid for debugging, syntax highlighting, wise code completion, snippets, code refactoring, and embedded Git [1].

The setup process of Visual Studio Code is as follows:

* + - * Open the official website of Visual Studio Code to download the software.

Fig 1.1 Downloading VS Code (*Source*:[1])

* + - Now select the version and OS you want to install for and click on download.
    - After the file downloads click on setup and then agree to the license agreements.
    - Select the destination folder for saving the VS code files and click on next.

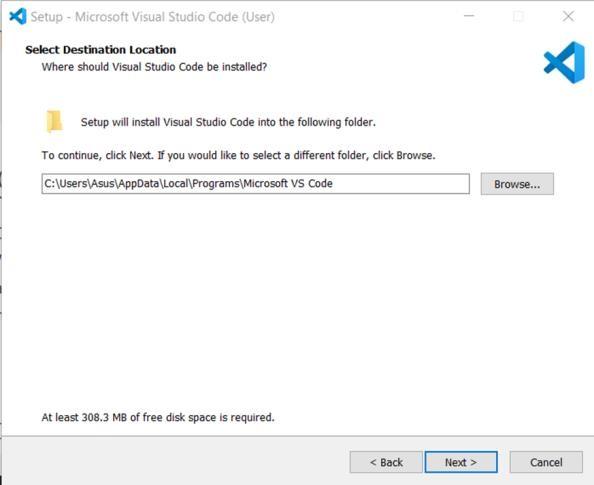
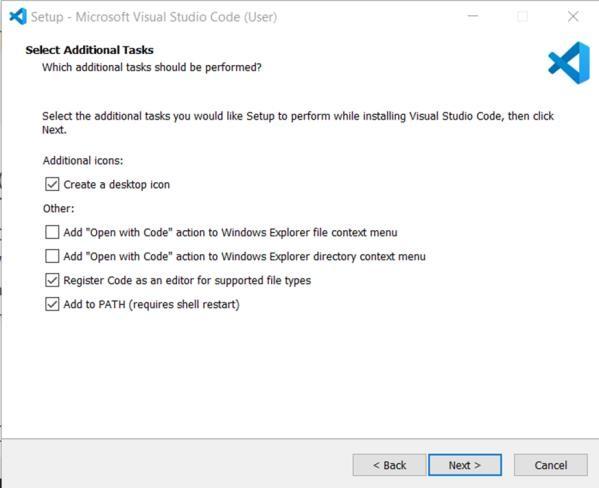
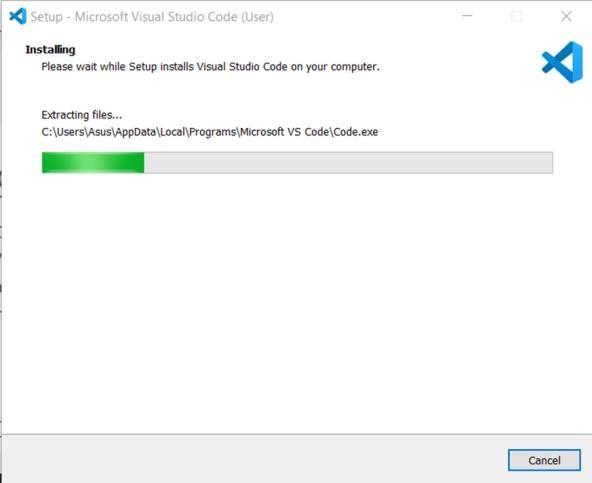


Fig 1.2 Installing VS Code (*Source*:[1])

* + - In the additional tasks window confirm that the “Add to PATH” is selected.

Fig 1.3 Configuring VS Code (*Source*:[1])

* Now complete the installation process by clicking on Install, it should take a few mins to finish installation.

Fig 1.4 Finishing VS Code Installation (*Source*:[1])

* + 1. **PyCharm Community Edition -** PyCharm is a committed Python Integrated Development Environment (IDE) supplying a huge variety of important equipment for Python developers, tightly included to create a handy surroundings for effective Python, web, and information technological know-how development [2].

The setting up of Pycharm is as follows:

● For downloading PyCharm go to the internet site https://www.jetbrains.com/pycharm/down load/ and

Click the “DOWNLOAD” hyperlink below the Community Section.

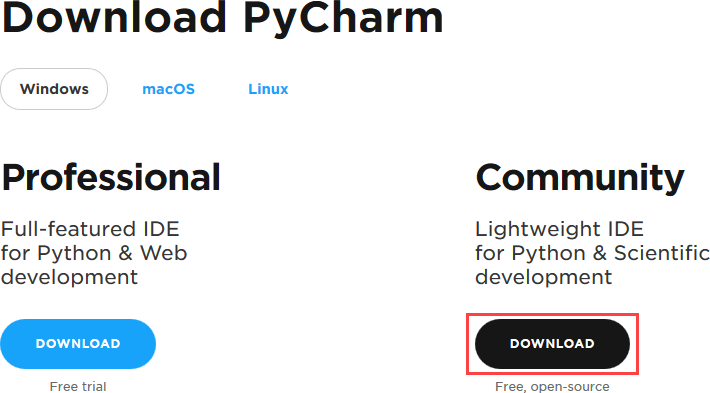


Fig 1.5 Downloading Pycharm IDE (*Source*:[2])

* + - * After downloading, run the setup to install PyCharm. Click “Next” on the setup Window.
      * Make a desktop shortcut if you want and click “Next”.

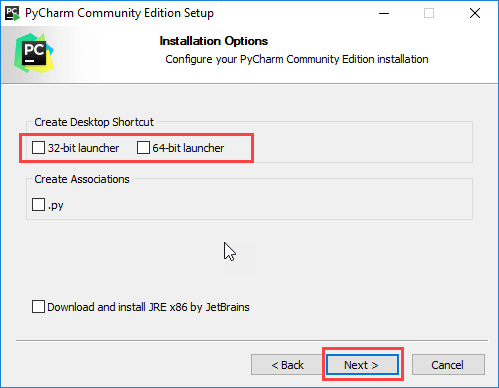


Fig 1.6 Configuring Pycharm (*Source*:[2])

* + - * On next window select Jetbrains and Install it.

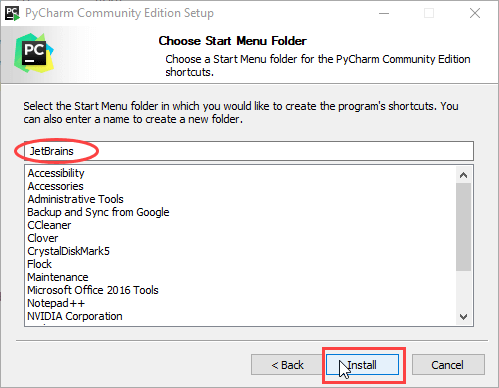
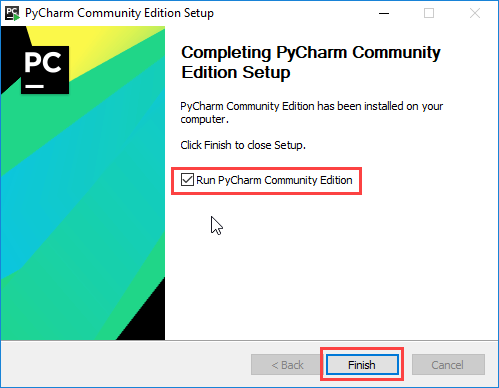


Fig 1.7 Following the installation steps (*Source*:[2])

* + - * The installation begins.
      * After it is done, you would get a message screen that PyCharm is configured. Click the “Run PyCharm Community Edition” box and finish the setup.

Fig 1.8 Pycharm installation complete (*Source*:[2])

* + - * The below screen will appear after the setup is finished.

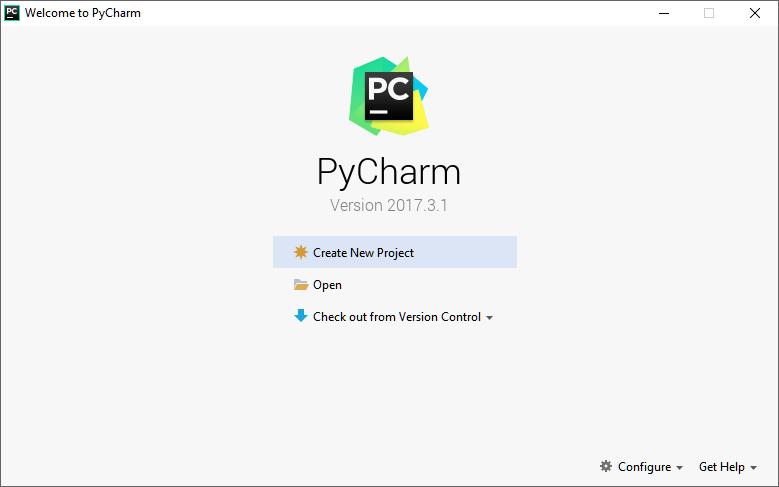


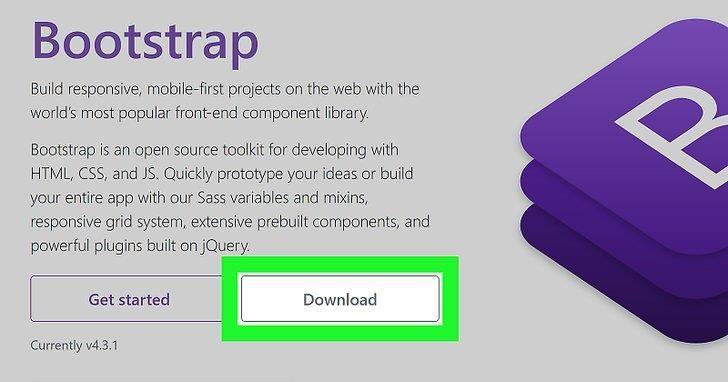
Fig 1.9 Pycharm shortcut key, desktop icon created (*Source*: [2])

* + 1. **Bootstrap Framework -** Bootstrap is a loose and open-supply CSS framework directed at responsive, mobile-first front-quit internet development. It includes HTML, CSS, and JavaScript-primarily based totally layout templates for typography, forms, buttons, navigation, and different interface components [3].

The setup procedure of BootStrap is as follows:

● Open the Bootstrap internet site on your net browser. Type https://getbootstrap.com into the cope with bar, and press Enter for your Keyboard.

● Click the Download button to open "Download" page

Fig 1.10 Downloading Bootstrap files (*Source*:[3])

* + - * Click the Download button to download the compiled CSS and JS files as a zip file.



Fig 1.11 Choosing version and downloading (*Source*:[3])

* + - * From the ZIP archive extract the files downloaded, and place it all in the same folder.

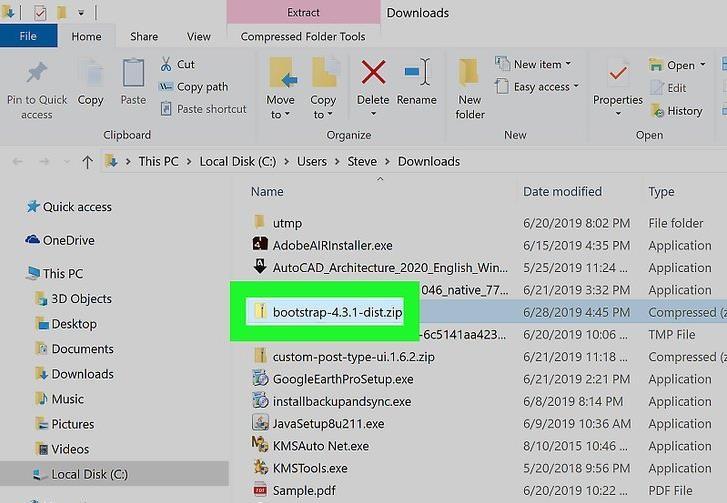


Fig 1.12 Extracting the downloaded file (*Source*:[3])

* + - * Move the extracted folders to the equal folder as your internet site HTML files. Open the folder that incorporates all of your internet site`s HTML files, and drag the "css" and "js" folders right here to transport them to the equal folder as your internet site documents.

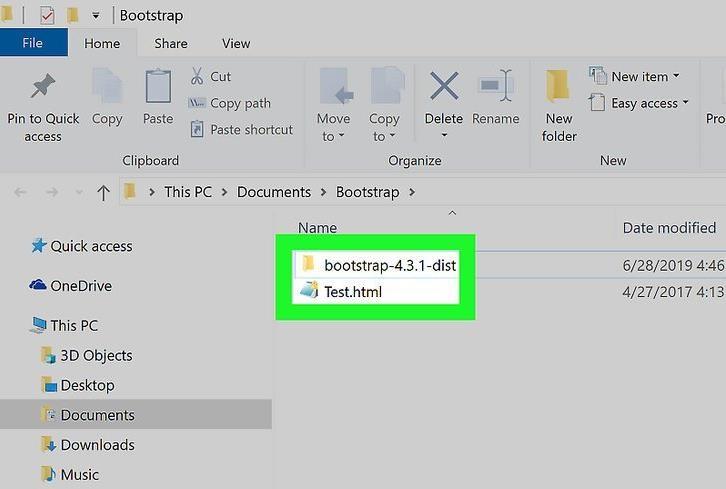


Fig 1.13 Move extracted files to project HTML folder (*Source*:[3])

* + - * Right-click the HTML file you want to use with Bootstrap. You can use Bootstrap in only one of your HTML files, or all of them.

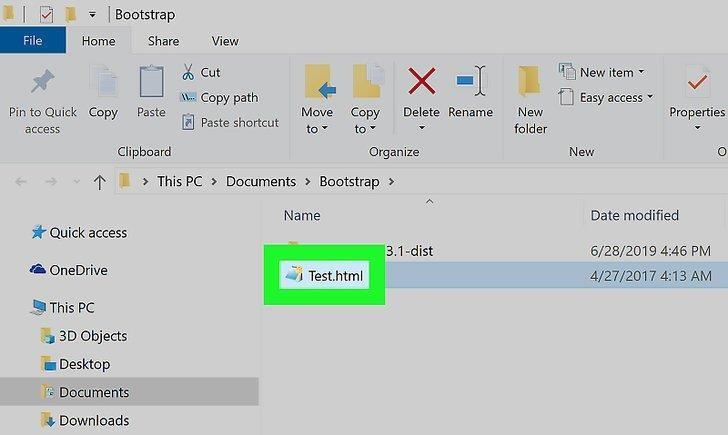


Fig 1.14 Adding Bootstrap to project files (1)(*Source*:[3])

* + - * A sub menu will pop up when you hover over Open with option.

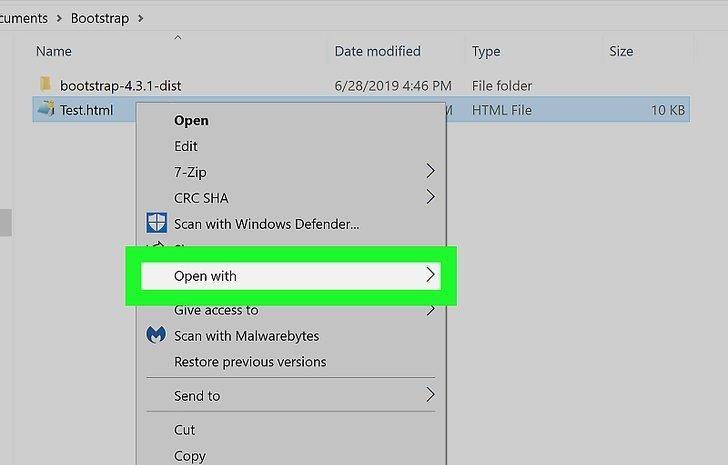


Fig 1.15 Adding Bootstrap to project files(2) (*Source*:[3])

* + - * Select your preferred text editor to open the Html file.

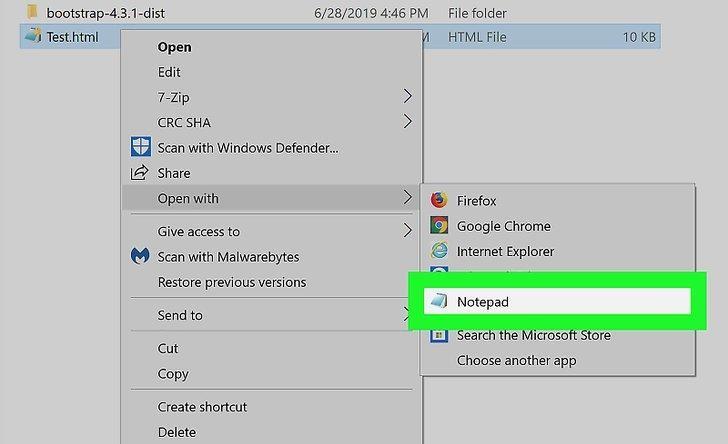


Fig 1.16 Adding Bootstrap to project files(3) (*Source*:[3])

* + - * In your HTML file’s header add the bootstrap links before using the code.



Fig 1.17 Adding Bootstrap to project files(4) (*Source*:[3])

* + - * The Links should be added between <meta> and <title> tags.

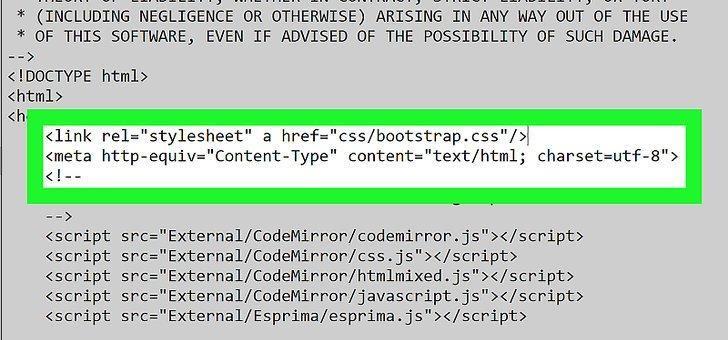
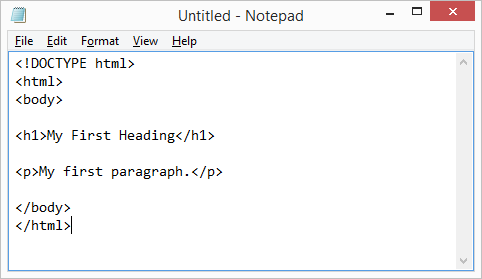


Fig 1.18 Adding Bootstrap to project files(5) (*Source*:[3])

* + 1. **Html -** HTML is short for Hyper Text Markup Language that is used for growing net pages and net applications. HTML is a language which is used for growing appealing net pages with the assist of styling, and which seems in a pleasing layout on an internet browser. An HTML record is fabricated from many HTML tags and every HTML tag includes extraordinary content [9].
       - Open Notepad and write the basic html file tags

Fig 1.19 Creating HTML file (*Source*:[9])

* + - * Save it on your PC as “index.html”.

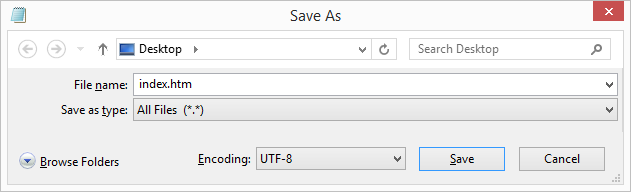


Fig 1.20 Saving File as HTML (*Source*:[9])

* + - * Double Click on the saved file to view it in preferred browser. It will look something like this:

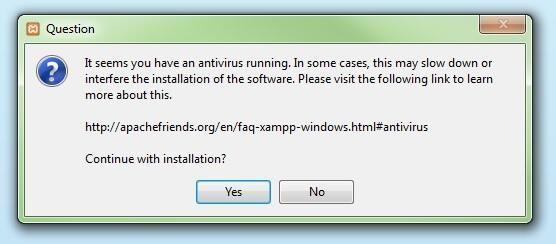


Fig 1.21 Opening HTML file in browser (*Source*:[9])

* + 1. **Xampp -** XAMPP is one of the broadly used cross-platform net servers that enables builders create and take a look at their packages on a nearby webserver. It become advanced through Apache Friends, and its local supply code may be revised or changed through the audience. It includes Apache HTTP Server, MariaDB, and an interpreter for the unique programming languages like PHP and Perl. It is to be had in eleven languages and supported through unique structures inclusive of the IA-32 bundle of Windows & x64 bundle of macOS and Linux [4].

The setup process of Xampp is as follows:

* + - * Download XAMPP from Apache Friends Website.
      * After downloading has finished click on the .exe to start installation.
      * It is recommended to pause any antivirus as it can hinder in the installation.

Fig 1.22 Setting up Xampp(1) (*Source*:[4])

* + - * You should also deactivate User Account Control (UAC) for the same reasons as it limits writing access in C drive.



Fig 1.23 Setting up Xampp (*Source*:[4])

* + - * You will see the start window of xampp. To configure the settings click on next.



Fig 1.24 Setting up(3) Xampp (*Source*:[4])

* + - * It is recommended to install using the default setup and available components. Click on next after choosing your options.

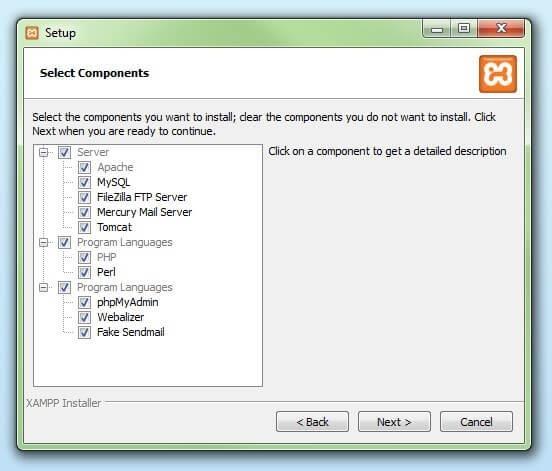


Fig 1.25 Setting up Xampp(4) (*Source*:[4])

* + - * Choose the location for the installation. Under the standard setup, a XAMPP in C:\ drive will be created. Click next after choosing location.

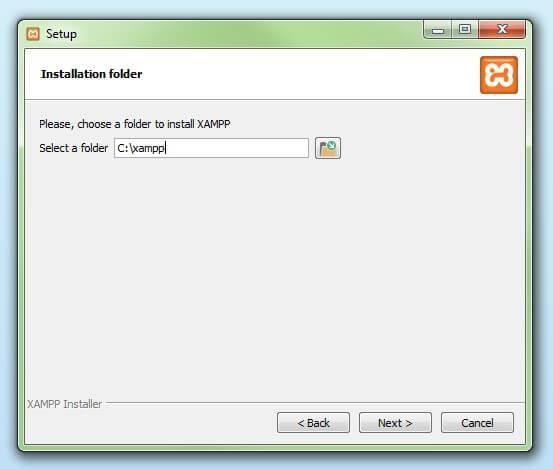


Fig 1.26 Setting up Xampp(5) (*Source*:[4])

* + - * After all the configurations have been done. The components will be installed and saved to the chosen directory. The green loading bar will show the progress on the screen.



Fig 1.27 Setting up Xampp(6) (*Source*:[4])

* + - * XAMPP installation may be interrupted by your firewall. To enable communication between the Apache server and your network use the check box.
      * To close wizard click on finish after all the configurations have been completed. For opening the XAMPP Control Panel click on the check box.



Fig 1.28 Setting up Xampp (7)(*Source*:[4])

* + 1. **PhpMyAdmin –** It is written in PHP, and is a free software tool supposed to deal with the management of MySQL over the Web. It helps a extensive variety of operations on various databases. Frequently used operations may be achieved thru the consumer interface, even as you continue to have the capacity to immediately execute any SQL statement [5].

Setup process for is as follows:

* + - * Go to the hyperlink [http://www.wampserver.com/en](https://www.wampserver.com/en/) the link can be used, or google it and the first link can be used to download it.
      * WAMP Server application can be downloaded from this website. The software details and features are mentioned on the site and the alias phpmyadmin tool.

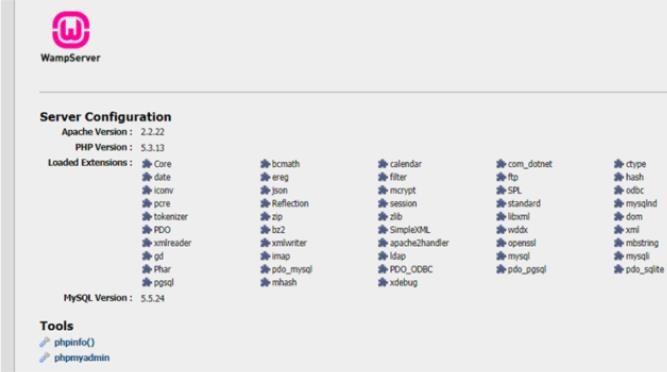


Fig 1.29 Setting up PHPMyAdmin (*Source*:[5])

* + - * After installation, it will show the commodities that come along with it like the Apache & localhost application, the databases like MySQL & MariaDB, phpMyAdmin & Adminer used for Administration purposes.



Fig 1.30 Viewing databases (*Source*:[5])

* + 1. **Python Programming Language -** Python is an object-oriented, interpreted, high-degree programming language with dynamic semantics. Its high-degree constructed in statistics structures, blended with dynamic typing and dynamic binding, make it very appealing for Rapid Application Development, in addition to to be used as a scripting or glue language to attach current additives together. Python`s simple, clean to examine syntax emphasizes clarity and consequently reduces the price of software maintenance. Python helps modules and packages, which inspires software modularity and code reuse. The Python interpreter and the vast well known library are to be had in supply or binary shape without fee for all most important platforms, and may be freely distributed [8].
       - Go to the official website and go downloads tab. As per your OS. Python3 is the latest version. Click on the download link as per your device specifications..



Fig 1.31 Downloading Python3 (*Source*:[8])

* + - * After downloading the installer, run it. You can check Add python to path check box as well as the checkbox for all launchers users.

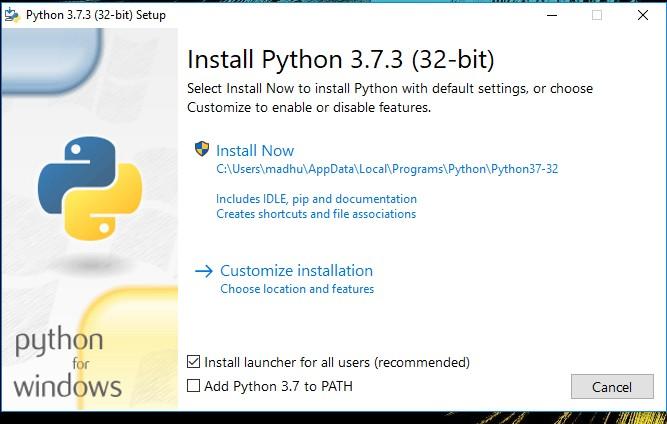


Fig 1.32 Installing Python3 (*Source*:[8])

* + - * Choose preferred installation. The optional features can be selected like [pip](https://www.digitalocean.com/community/tutorials/python-pip), documentation, Idle and tcl/tk. To make it easier to start python install the global launcher for .py files.

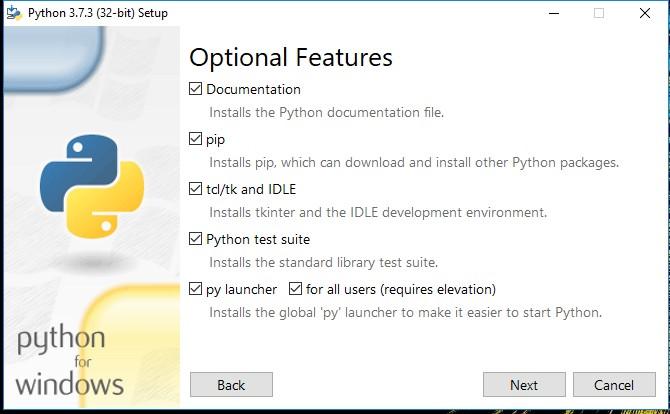


Fig 1.33 Configuring setup (1)(*Source*:[8])

* + - * To go to advanced options click on next. Select the checkboxes for path and all users. Additional Options are also provided for different purposes. Note down the python installation directory. After Advanced options are configured, to start the install click on install.

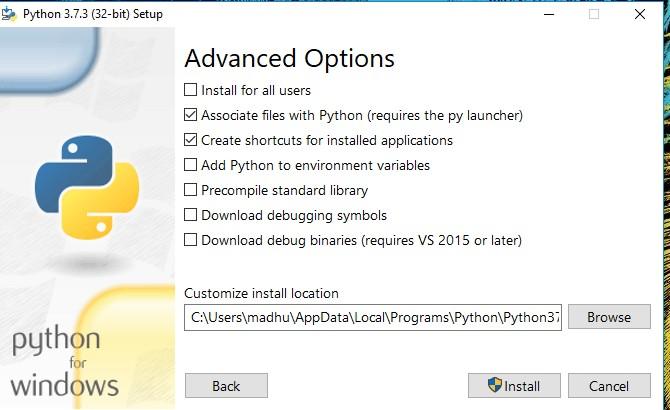


Fig 1.34 Configuring setup (2) (*Source*:[8])

* + - * After setting up you will see a setup successful message.

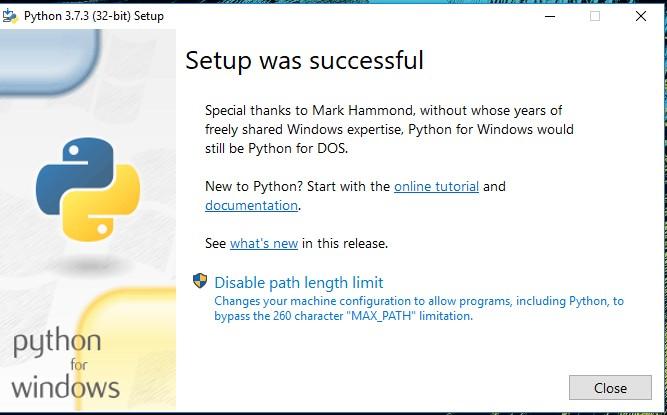


Fig 1.35 Finishing Python installation (*Source*:[8])

* + 1. **Flask** – It is written in python and is a web application framework. Armin Ronacher was the one responsible for its development, he was the leader of Poocco a team of international python fanatics. It is based on the Jinja2 template engine and Werkzeg toolkit [11].
       - It is a python library which can be installed from the command prompt.
       - Open the command prompt and type pip install flask.
       - This will install all the flask libraries on your system.
       - Alternatively, you can go under the packages section of your favorite IDE and install the flask package. Now you can use flask in your python projects.
  1. **Overview of the report**

This document is a comprehensive report on the existing college management systems. It takes a short glance at their features as well as limitations. Additionally, it includes a catalogue of attributes that will be incorporated in our final product along with the construction steps, procedure, execution and culmination.

**Chapter 2**

**System Analysis and Design**

1. **H**
   1. **Requirement Specifications**

This project “Kampus Kit” is a web application and therefore can be run on any device as well as a browser. It is recommended to use the latest versions of the browsers to avoid and crashes. This web application does not require the resources of the host machine for any sort of database storing or for performing any computing tasks.

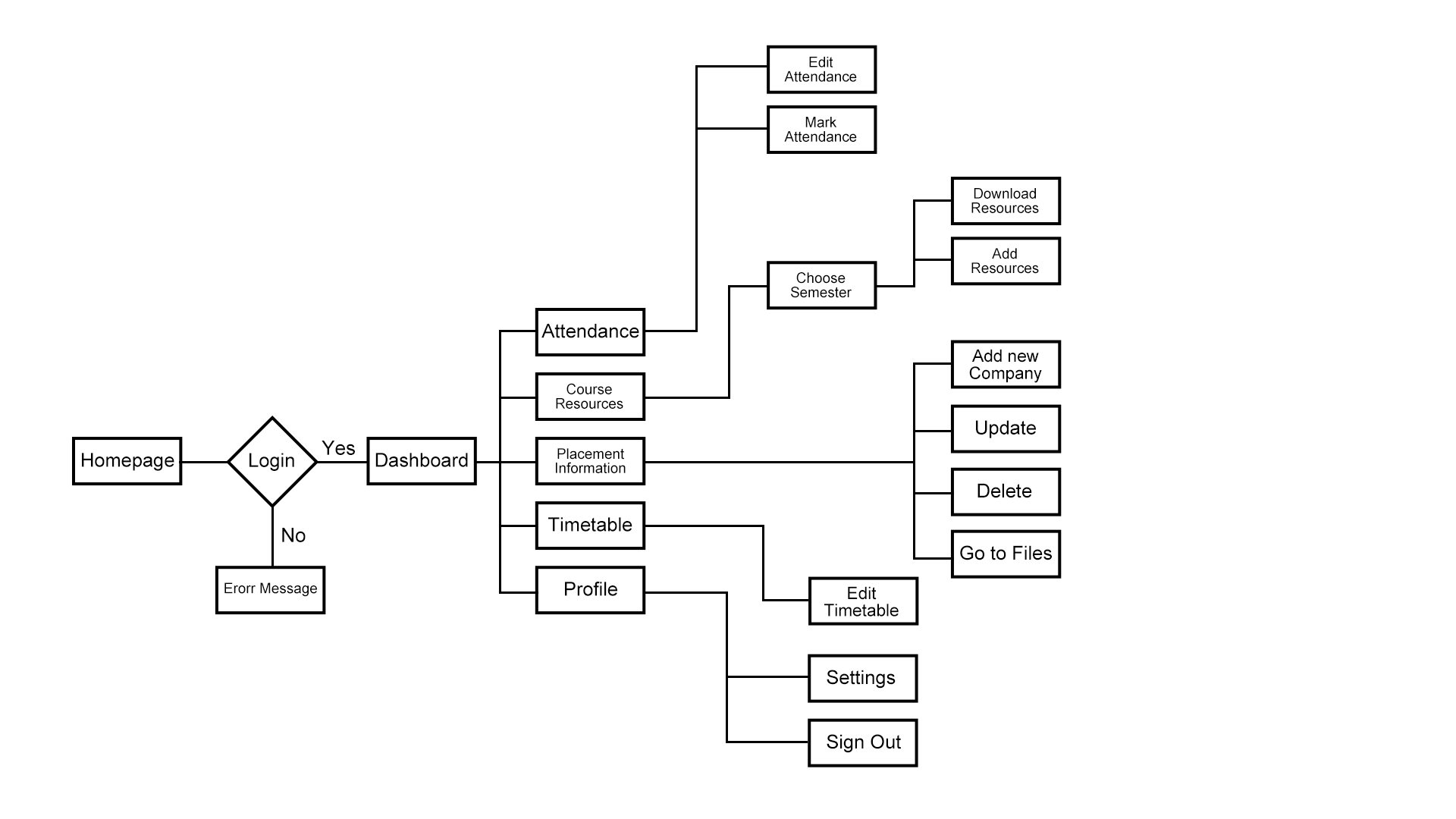
* 1. **Flowcharts**

Given below is a visual representation of the workflow of the web app as well as how the data is acquired:

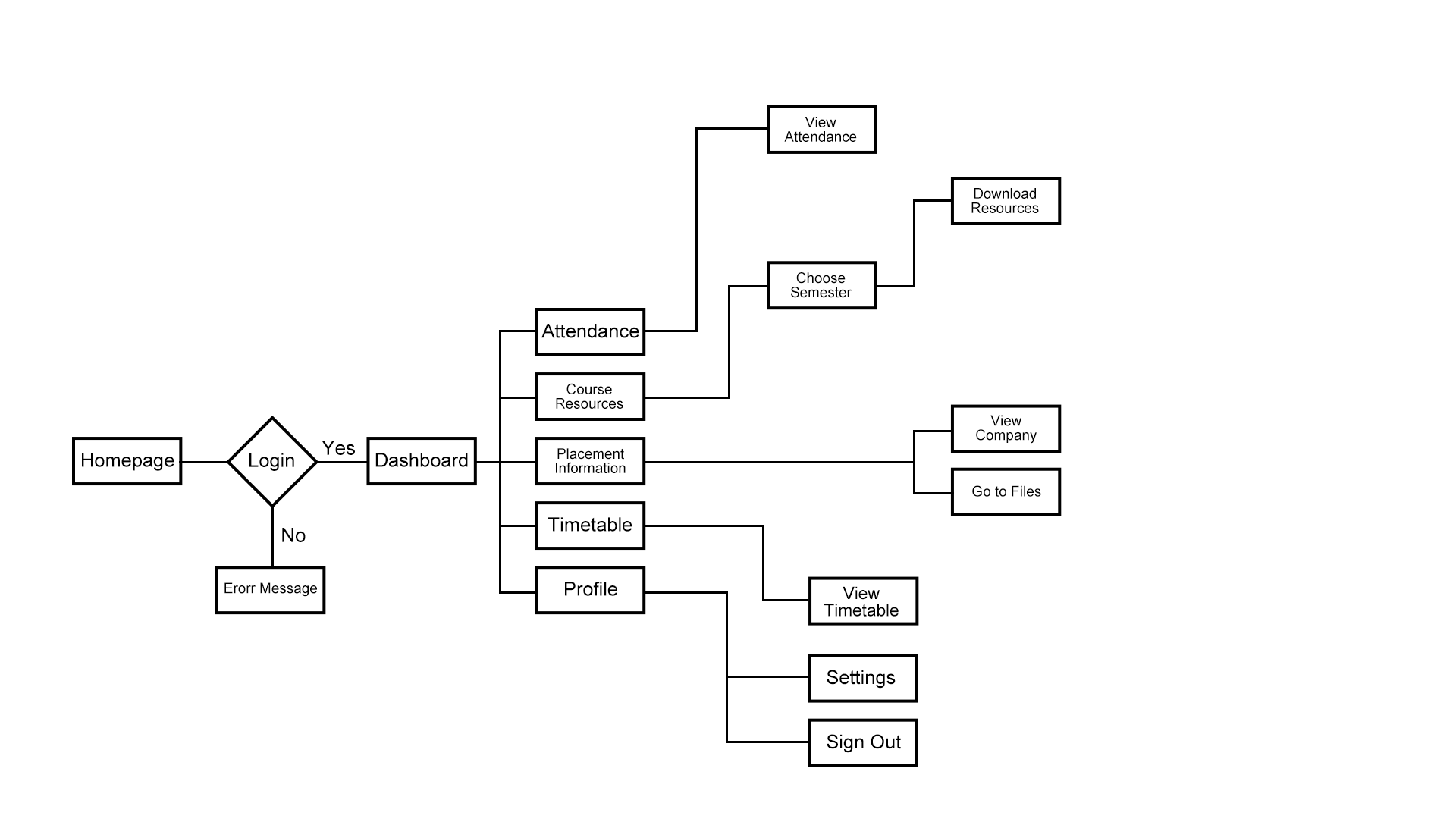
In Fig 2.1 the admin workflow of the application can be seen, any user irrespective of the their authorization level will be directed to the homepage of the application initially. It is on logging in that the separation between an admin and a user can be seen. The available options to the admin and a normal user differ heavily.

Fig 2.2 shows the user workflow of the application. Just like with the admin workflow, everyone gets directed to the homepage initially. After logging in the dashboard that is visible to a normal user mostly comprises of only the ability to view contents and information.

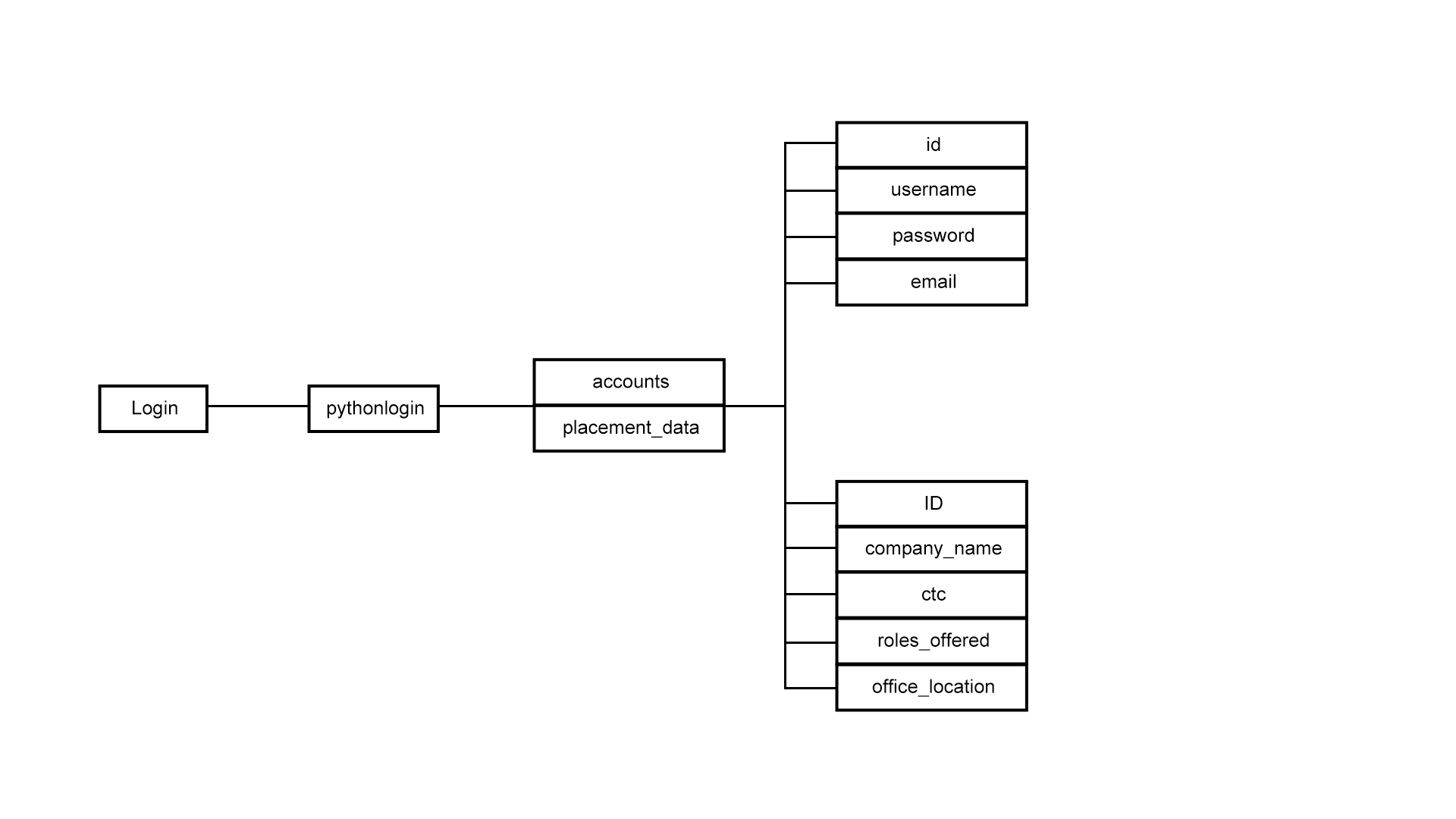
In Fig 2.3 the tables inside the database along with attributes of the tables can be seen. When a user logs into their account, their info is called from the accounts table and is verified if the entered information matches the stored one or not. The second table placement\_data comes into play only after a successful login, that is when the user opens the placement information tab in the dashboard.



**Fig 2.1** Admin Flowchart (*Source*:[Self]) - this figure comprises of all the pages of the application while also showing the connction between them for an admin.



**Fig 2.2** User Flowchart (*Source*:[Self]) - this figure comprises of all the pages of the application while also showing the connction between them for a user.



**Fig 2.3** Database Flowchart (*Source*:[Self])- this figure comprises of the tables and their attributes inside the database.

* 1. **Design Steps**

In order to fulfill the objectives proposed at the early stages of the project each step of the designing process needs to be meticulously carried out to avoid any issues in the later stages.

The steps carried out include:

* Coming up with an appropriate project title that can convey the primary objective of the application and sounds aesthetically pleasing.
* Researching similar products and solutions offered by other companies and listing out their pros and cons.
* Coming up with a basic list of features that need to be implemented in the prototype.
* Creating the wireframe and then the prototype of the web app using designing softwares like Adobe Photoshop and Figma.
* Writing down the existing softwares and technologies that can be used to build the app according to the desired specifications and ideas.
* Finalising the technology to be used after discussion and beginning the process of building web pages.
* Building the backend of the web application on Python using Flask framework while simultaneously working on the web pages using bootstrap templates.
* Connecting the web pages using endpoints in Flask and adding an authentication system.
* Adding a database to the authentication system to ensure only authorized users have access to the resources.
* Fixing bugs and improving UI based on feedback from users, furthermore adding new features to the web application.



**Fig 2.4** Kampus Kit Logo(*Source*:[Self]) application logo created using Figma.

* 1. **Pseudo Code**

This section will comprise of the pseudocode for the main backend file being used:

* Import the libraries Flask, render\_template, session,redirect, url\_for, MySQL, MYSQL.cursors, re, SQLAlchemy, Usermixin, login\_user, LoginManager, Login\_required, Logout\_user,current\_user, FlaskForm, StringField, PasswordField, SubmitField, InputRequired, Length, ValidationError, Bcrypt, Mail, json, smtplib, datetime, session, subprocess
* Next open the configkk1.json file as c and load the parameters
* params=json.load(c)["params"]
* Set the Secret key to login
* app.secret\_key="login"
* Set the host,user,password, and database Paramaters.
* Define the default route using home function and render the index template.
* Define the route to login\_kk using the login function
* Set message to an emptty string msg = ''
* Get the email and password from the login form using the GET/POST method and store it in variables.
* Define a cursor using mysql connection.
* Using the select mysql query select from accounts table where email=email and password=password.
* Store it in account variable using fetchone fucntion of the cursor and close the cursor.
* If there is a value in account set the session variables loggedin,id and email and redirect to dashboard.
* Else display a message of incorrect email/password and render the login page while passing the msg variable to the template.
* Define the route to login\_kk/dashboard using the dashboard() function that renders the dashboard\_index template.
* Define the route to daywise\_attendance using daywise\_attendance() function that renders daywise\_attendance2 template.
* Define the route to course resources using course\_resources() function that renders course\_resources2 template.
* Define the route to sem1 using sem1() function that renders sem1 template.
* Define the route to sem2 using sem2() function that renders sem2 template.
* Define the route to sem7 using sem7() function that renders sem7 template.
* Define the route to placement info using placement info function that renders placement\_info template.
* Define a Cursor using mysqlconnection.
* Using the select mysql query select all the fields from placement data.
* Close the cursor and render the placement\_info template.
* Define the route to add using add() function.
* Recieve the form values using get/post method and store them in variables.
* Create a Cursor using mysql connection.
* Using the insert mysql query enter the values stored in variables to the placement\_data table.
* Commit the changes close the Cursor and redirect to placement\_info template.
* Define the route to update using update(id) function and recieve the value of id from placement\_info template.
* Recieve the form values using get/post method and store them in variables.
* cn=request.form['company\_name']
* ctc=request.form['ctc']
* roff=request.form['roles\_offered']
* loc=request.form['office\_location']
* Create a Cursor using mysql connection.
* Using the update mysql query update the values stored in placement\_data table where ID=id.
* Commit the changes close the Cursor and redirect to placement\_info template.
* Define the route to delete using delete(id) function and recieve the value of id from placement\_info template.
* Create a Cursor using mysql connection.
* Using the delete mysql query delete the values stored in placement\_data table where ID=id.
* Commit the changes close the Cursor and redirect to placement\_info template.
* Define a route login\_kk/logout using the logout function.
* Pop the session variables loggedin,id and email and redirect to login template.
* Run the App in Debug Mode.

1. **F**

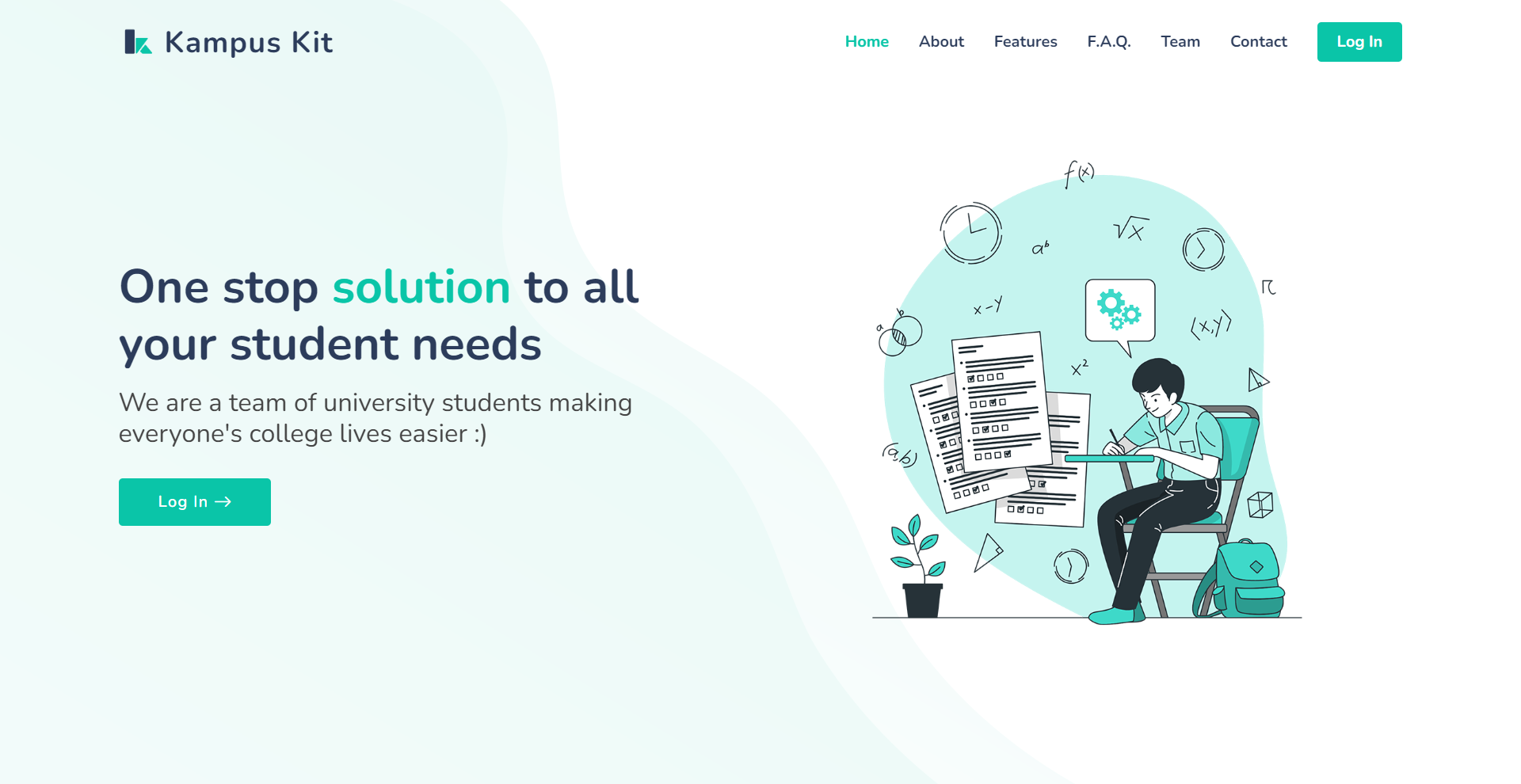
**Chapter 3**

**Implementation and Results**

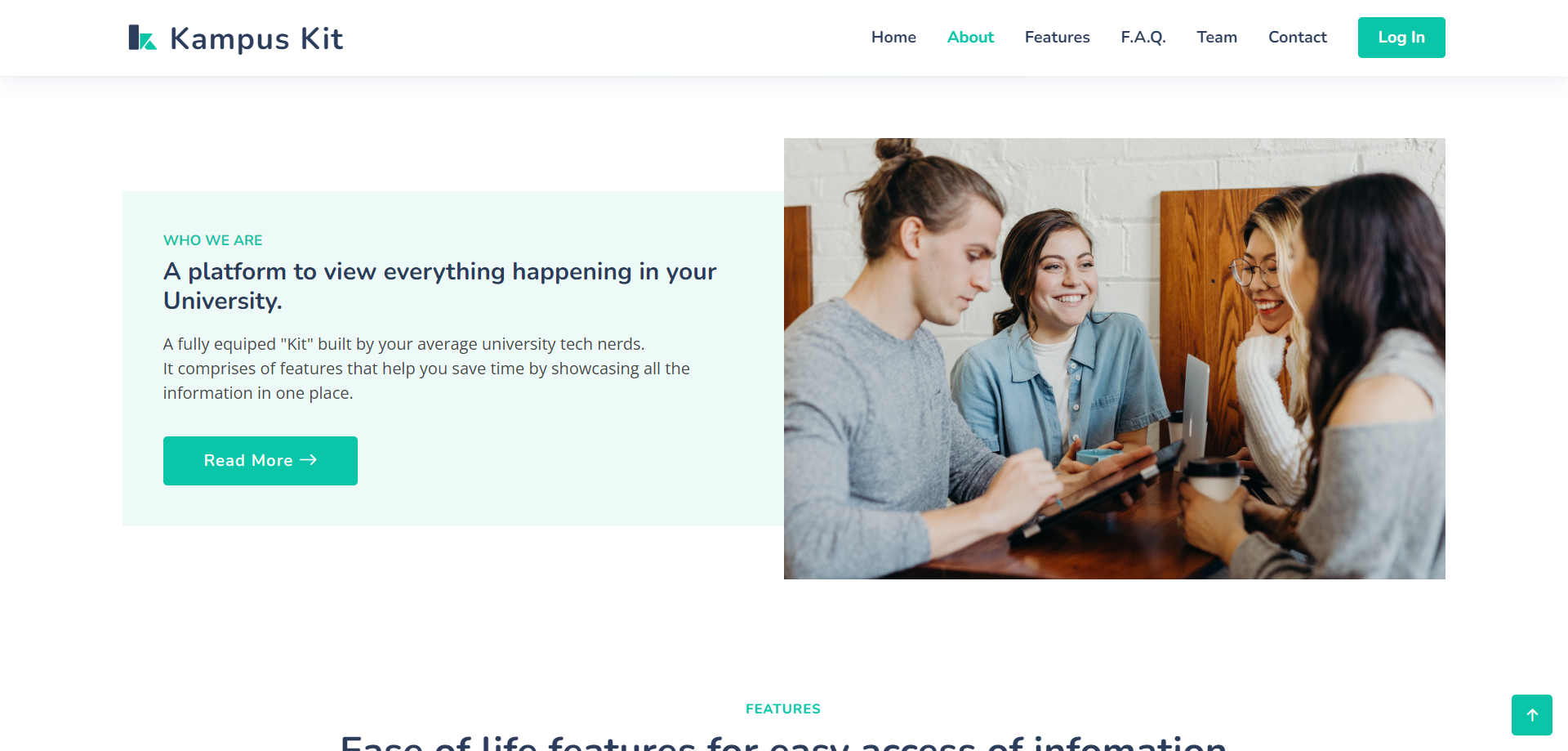
1. **F**
   1. **Main Python File** - This is the main backend file (kk1.py) that is responsible for connecting all the front end pages and for connecting the authentication system to the database. It is written in Python3 programming language and uses Flask framework to create and run the web application. Furthermore it uses multiple libraries for functions like automated mailing, connecting to database and its management, authentication, defining end points etc.

Here is the code file for the kk1.py:

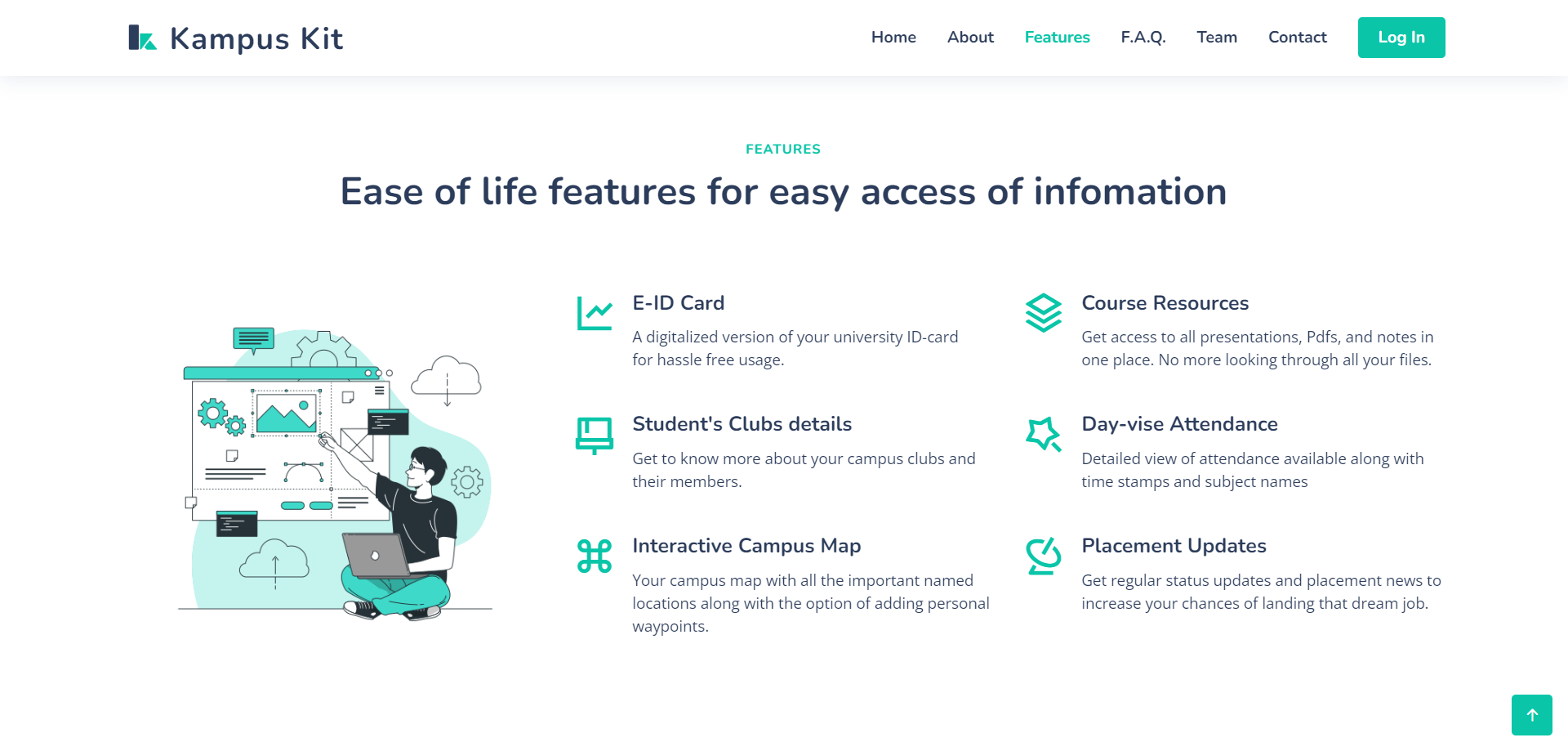
* 1. **Homepage** - This is the homepage (index.html) of the application that is visible to everyone who visits the web app. It comprises of a brief description of “Kampus Kit”



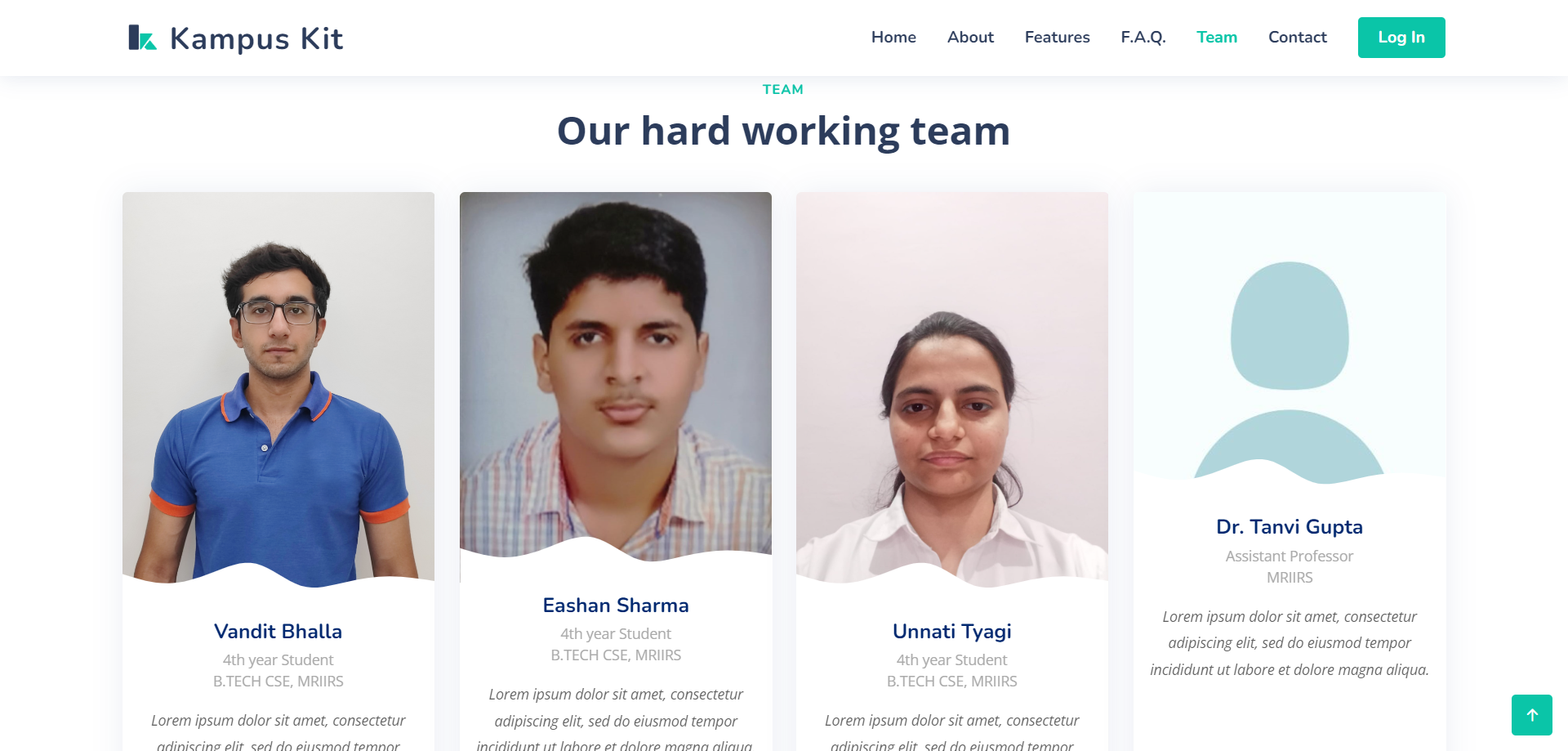
**Fig 3.1** Homepage(1) (*Source*:[Self]) the portion of the homepage that is visible to everyone on opening the application can be seen.



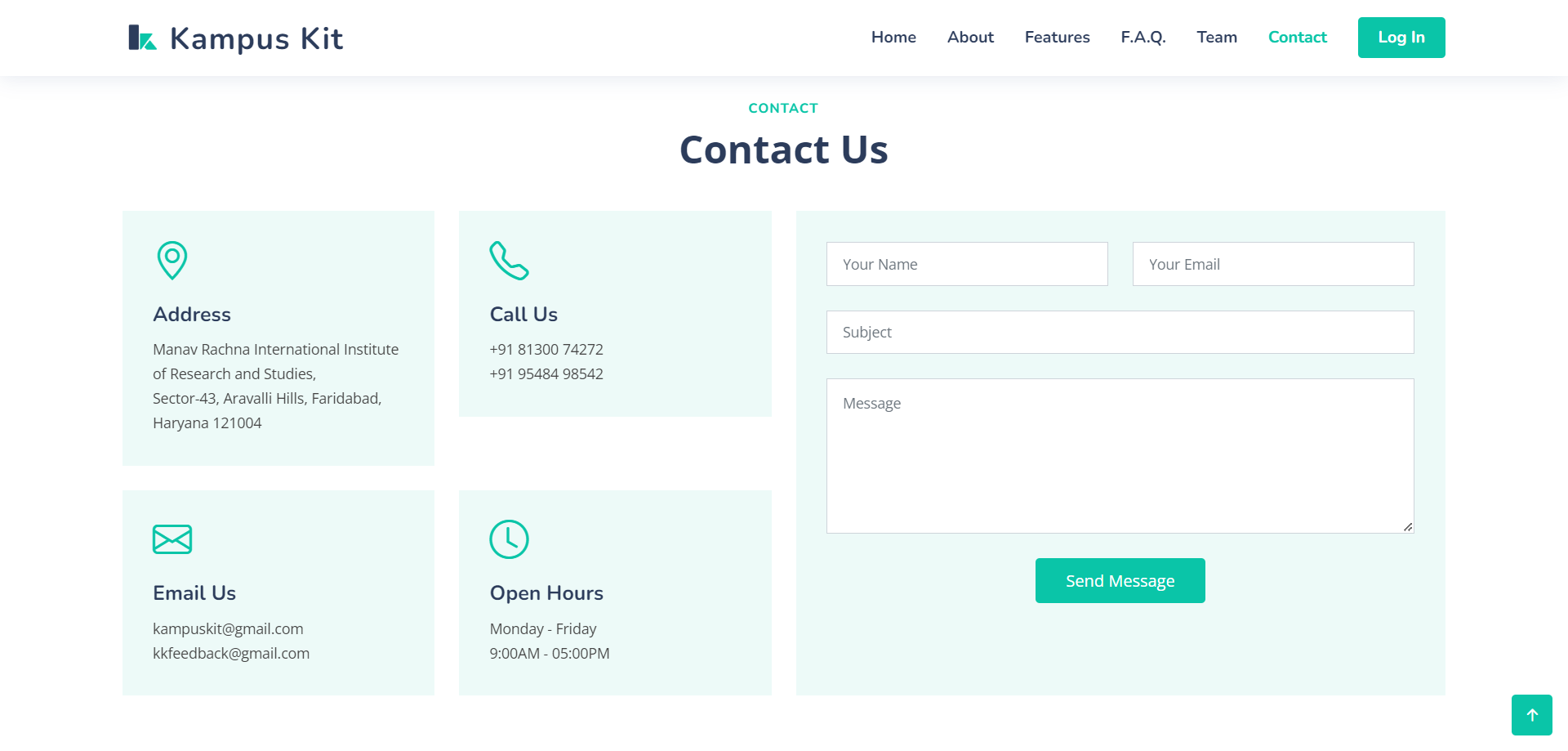
**Fig 3.2** Homepage (2) (*Source*:[Self]) shows the second section of the homepage which comprises of a brief description of the application.



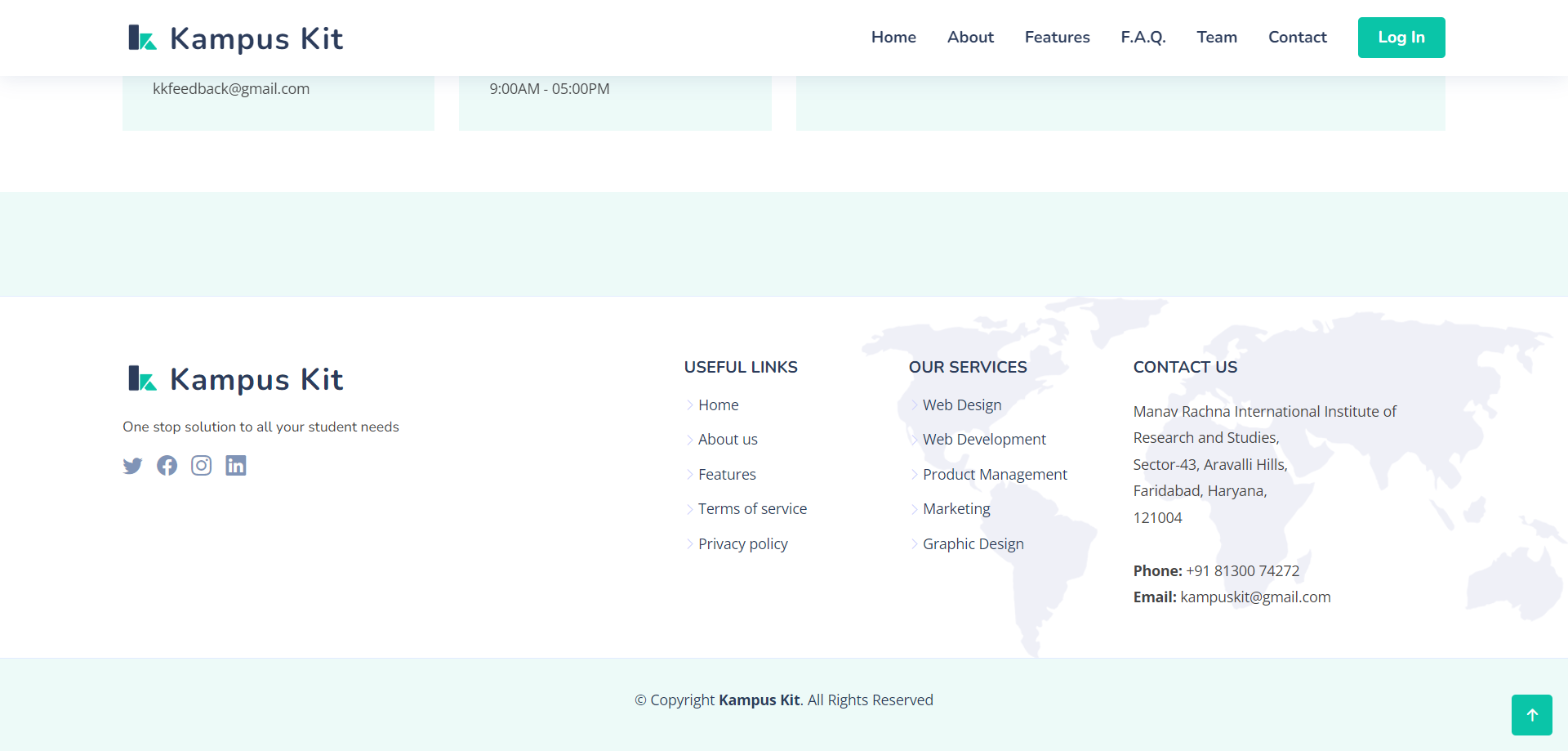
**Fig 3.3** Homepage (3)(*Source*:[Self]) shows the third section of the homepage where all the unique features of the application can be seen.



**Fig 3.4** Homepage (4)(*Source*:[Self]) this section comprises of the team members that have built the application along with who they are.

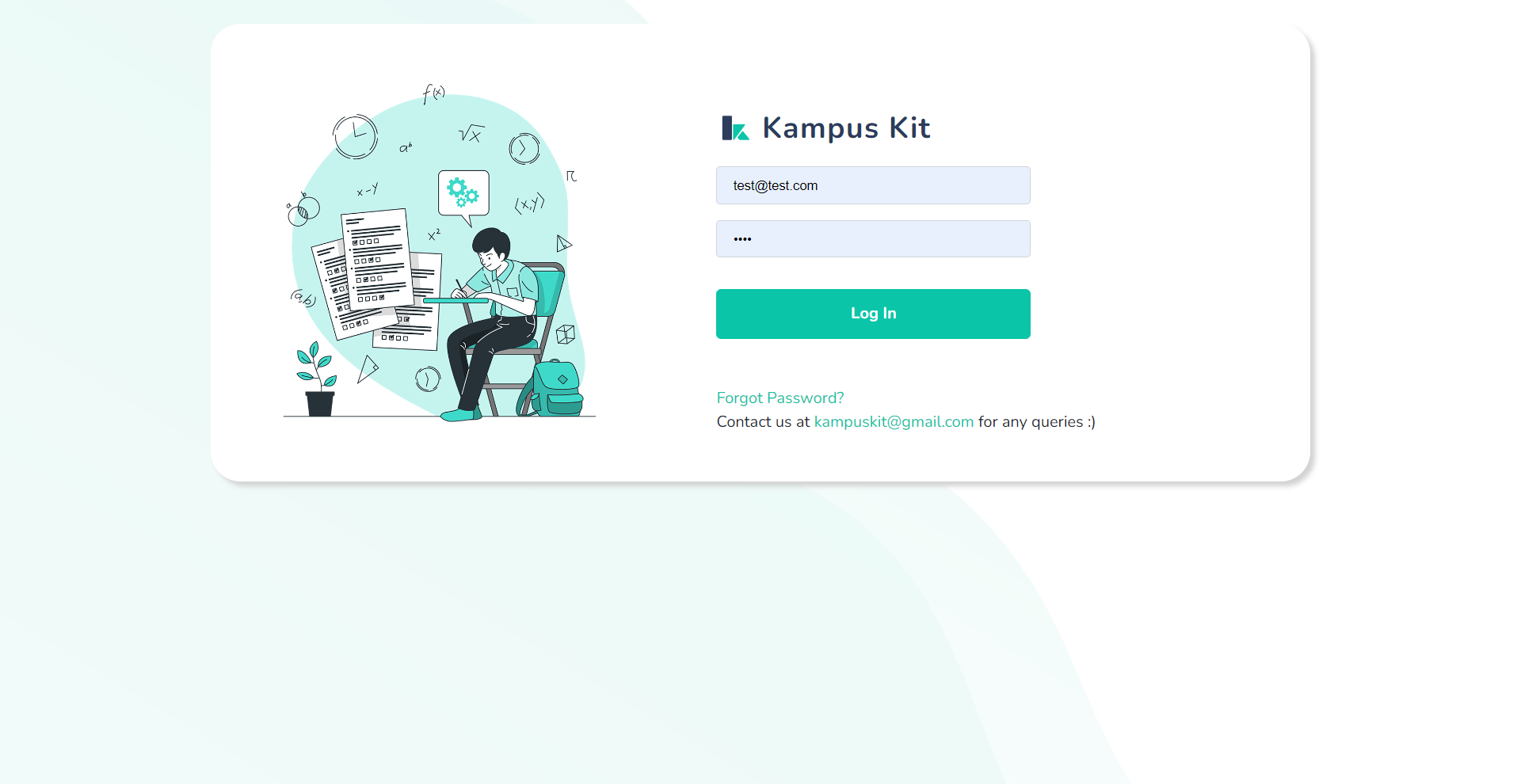


**Fig 3.5** Homepage (5)(*Source*:[Self]) in this section there is a form available for contacting in case of some queries or in case of some issue.



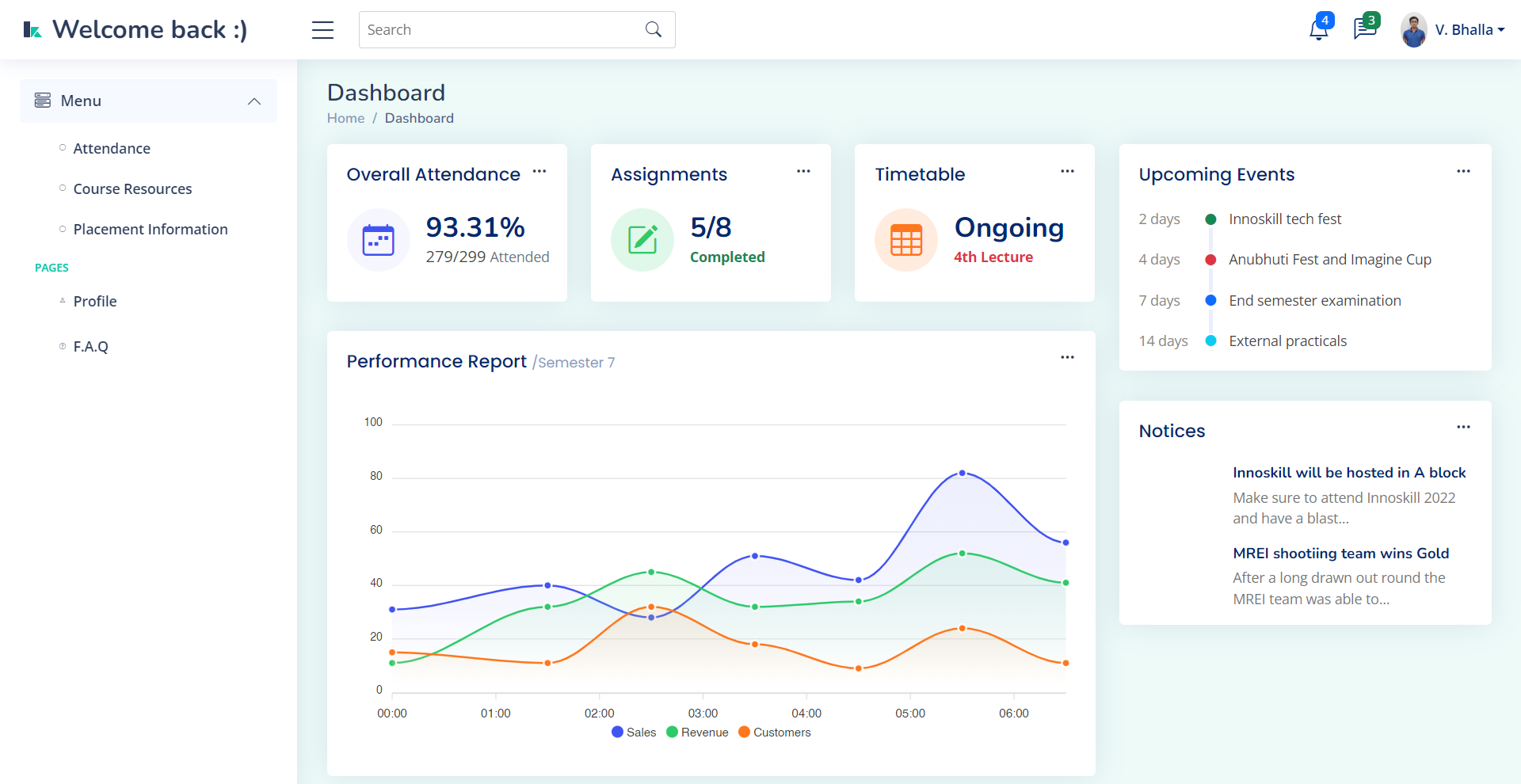
**Fig 3.6** Homepage (6)(*Source*:[Self]) the last section of the homepage holds the important links related to the application like social media links, privacy policy, contact details etc.

* 1. **Login Page** - This page (login\_page.html) is very crucial as this is where the user selects a choice between admin, faculty, and student. Depending on the choice the dashboard will open. In case the credentials entered are incorrect then it will display the message “Incorrect username/password”.



**Fig 3.7** Login Page (*Source*:[Self]) - this figure show what the login page of the application looks like. Only registered users will be able to pass the authentication.

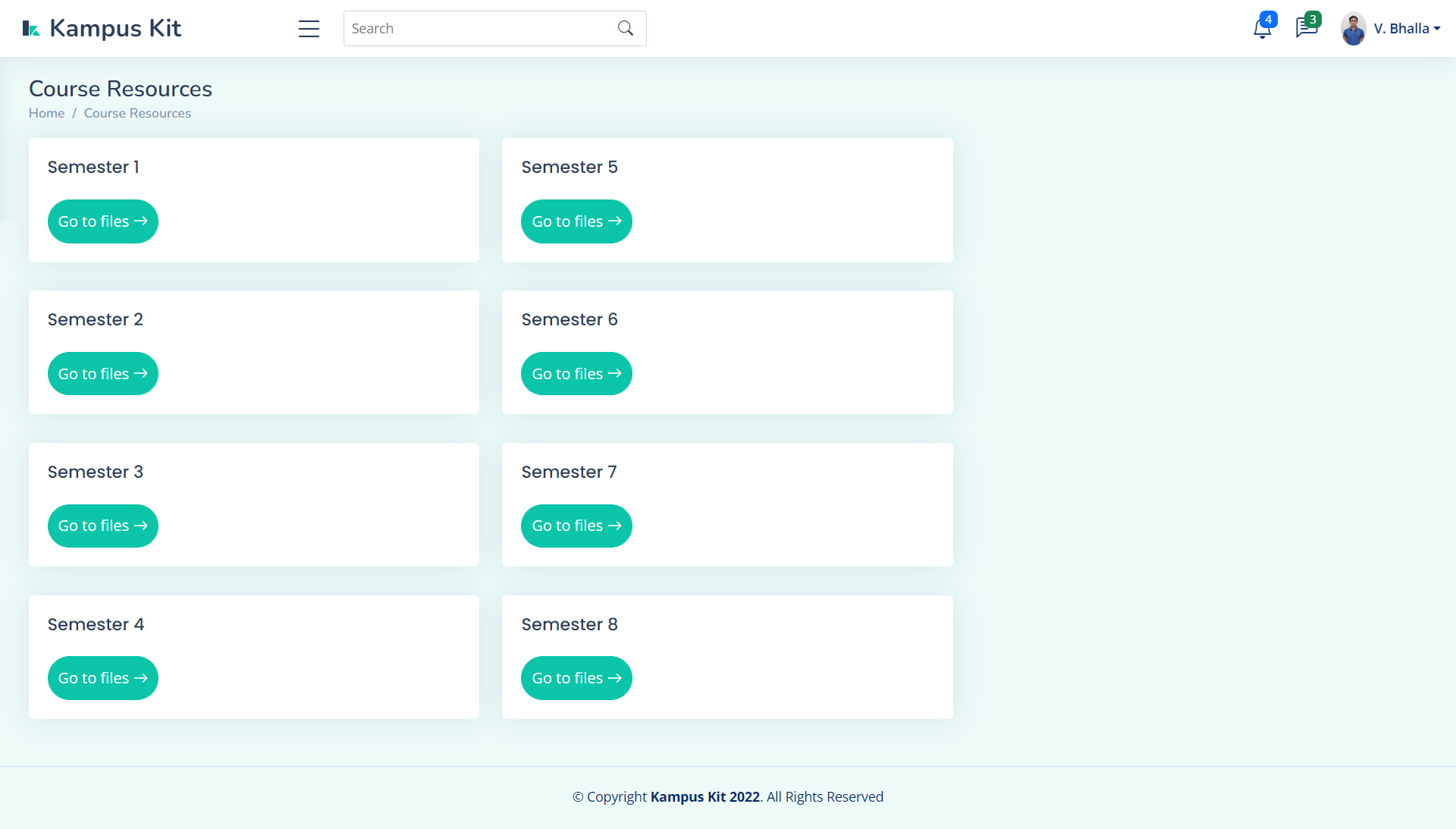
* 1. **Dashboard Page** - Once the user logs in they will get to see a dashboard (dashboard\_index.html) according to who they are and what access level they have. Here you can view all the features of Kampus Kit.



**Fig 3.8** Dashboard (*Source*:[Self]) - after the authentication process the users will be redirected to their respective dashboards depending on their role.

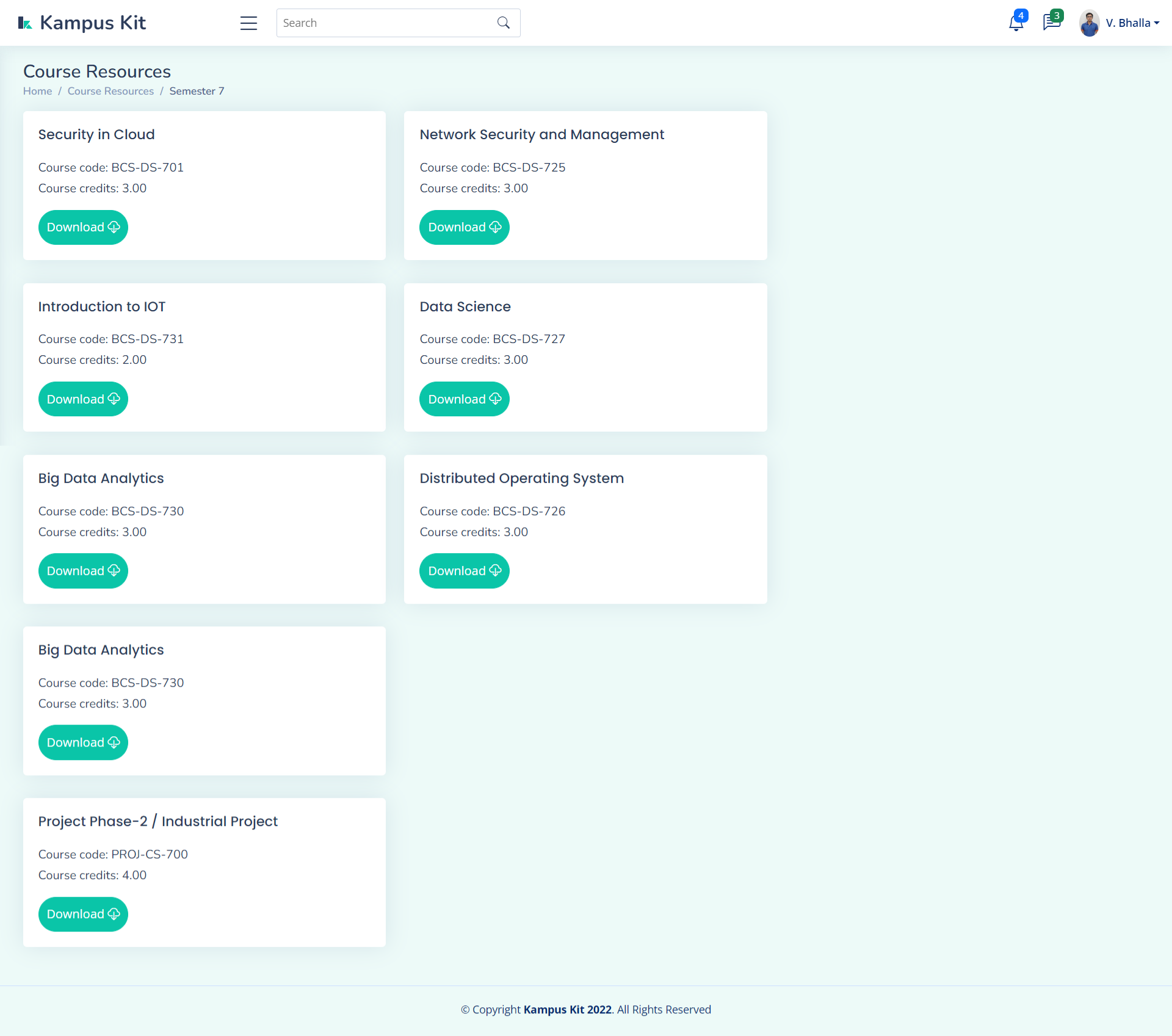
* 1. **Course Resources Page** - This page (course\_resources2.html) shows the user details about the courses or subjects that they are studying and along with it the user can also download the course material like presentations and notes for the same.

Here is the code file for course\_resources2.html:



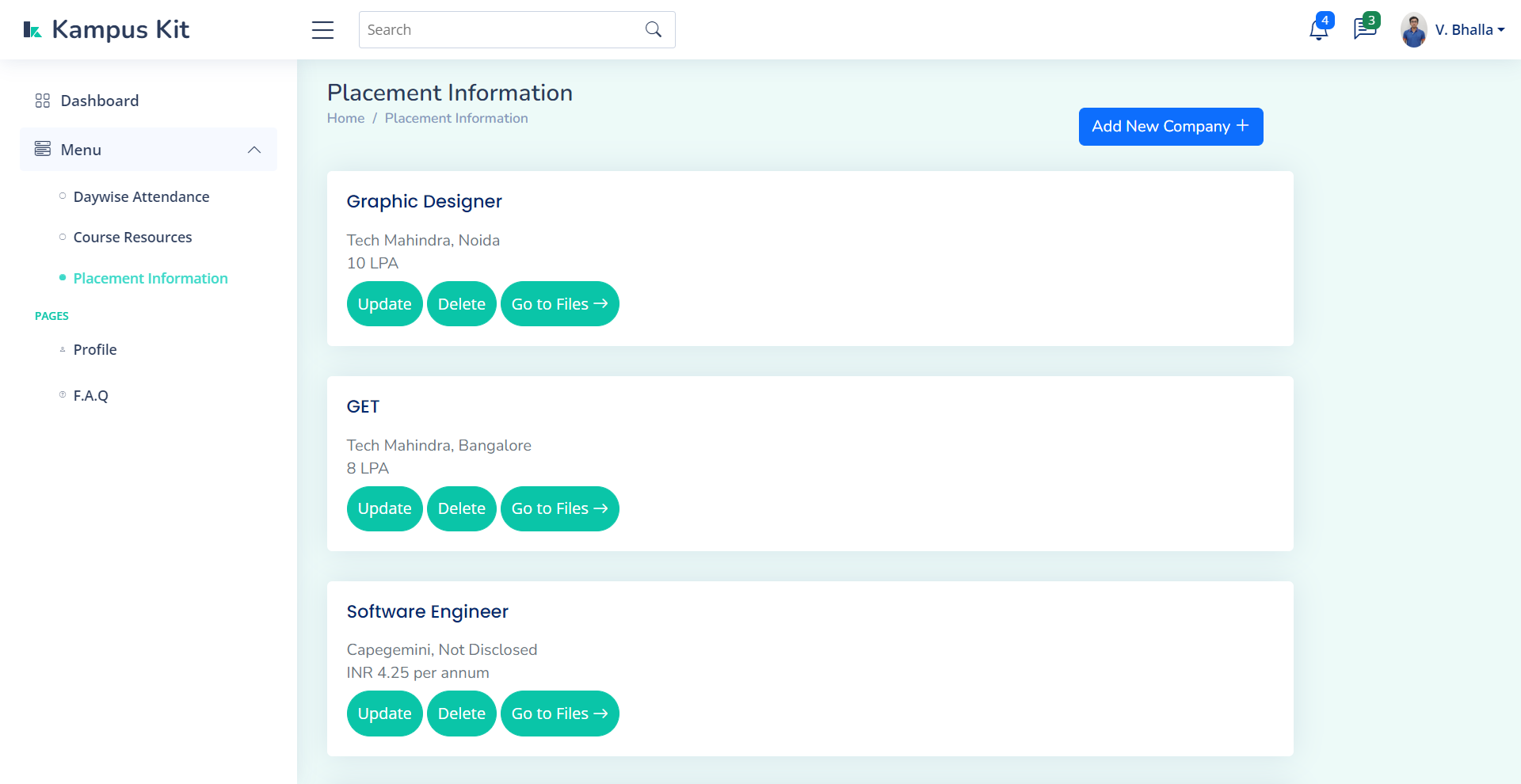
**Fig 3.9** Course resources(1) (*Source*:[Self]) - this figure comprises of a list of all the semesters that a student has.

* 1. **Semester Page** - This is what an individual semester page (sem7.html) will look like. It will have all the subjects that the student is enrolled into along with information about them and an option to download course material.

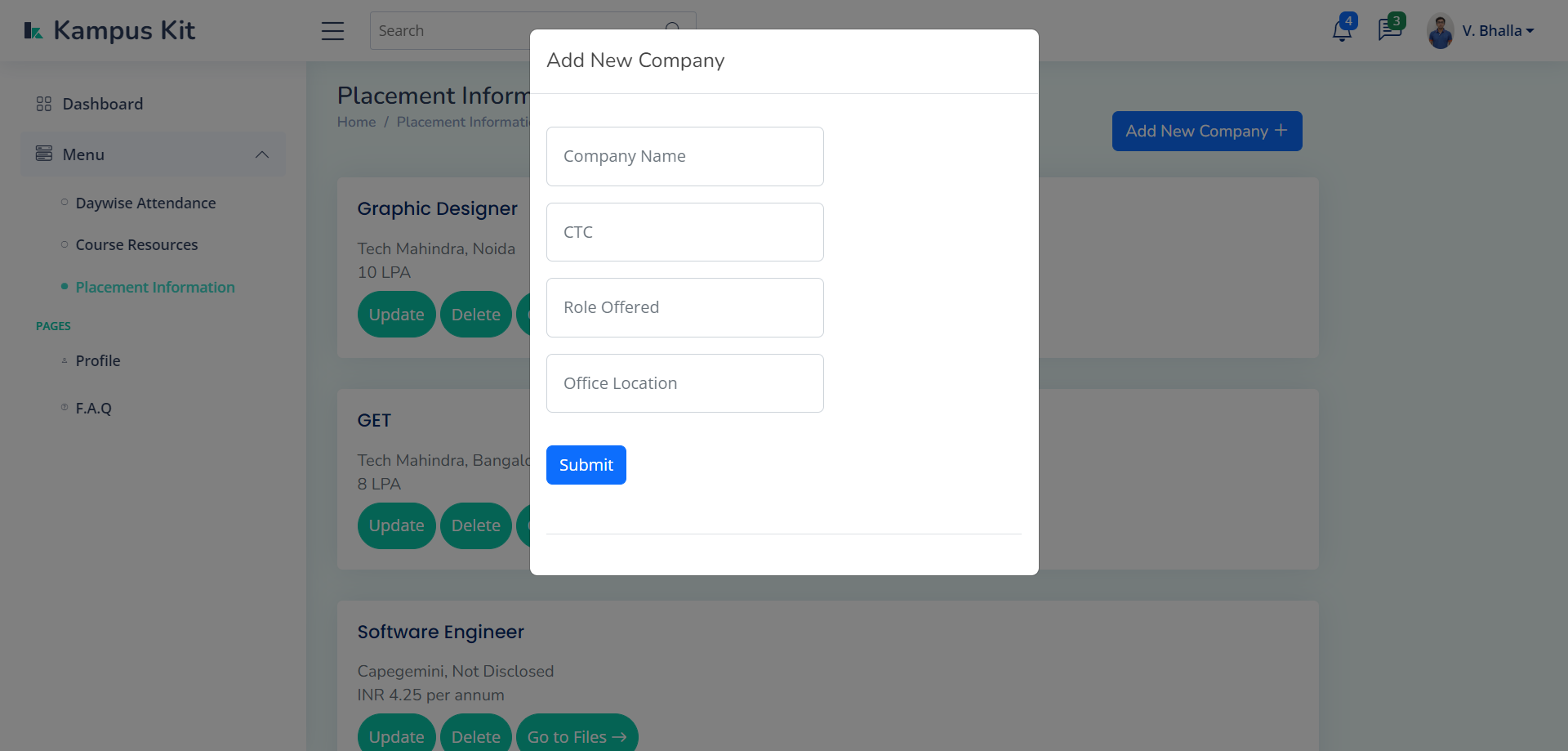


**Fig 3.10** Course resources(2) (*Source*:[Self]) - in this figure the course details along with their course code, credits and an option to download the resource files is available.

* 1. **Placement Page** - This page (placement\_info.html) will hold all the information regarding the placement drives that are being conducted on the campus for the final year students. In the admin version of the page ther will be options like Add, Update, Delete, Upload, whereas in the student version of the page there would be only two options Download and Apply.



**Fig 3.11** Placement Info(1) (*Source*:[Self]) - this figure shows a list of all the companies that have come for recruitment along with the job description and the CTC.



**Fig 3.12** Placement Info (2)(*Source*:[Self]) - the admin or the person incharge of handling company information can add the details of a new company.



**Fig 3.13** Placement Info(3) (*Source*:[Self]) - the admin can even update the information of existing companies incase of any change in the job description, dates or the package offered.

* 1. **Style Files(CSS)** - These two files (style.css and style-dashboard.css) are the CSS style files that are used to store styles for the various classes in the above mentioned HTML pages. As the name suggests style.css is for the homepage whereas the style-dashboard.css is for the dashboard.
     1. Homepage -
     2. G

**Chapter 4**

**Conclusion and Future Enhancements**

1. **F**
2. **F**
3. **F**
4. **F**
   1. **Summary of Work done**

At this point of time we have created a functional ERP inspired web application that conatins the following:

* A homepage which contains our brand logo of Kampuskit,Our tagline,the list of available Features,a FAQ Section, Information of the hardworking team and a contactUs Form.
* The Authorized users login to the Kampus Kit Web app using the login Page, if the user is genuine then he is taken to the user Dash board a message pops up conveying that the credentials entered are invalid.
* Dashboard contains a drop down menu listting the set of avialable features, a notice tab, an upcoming events tab and Performance Graph which shows the performance of a student.
* All the corporations that came and are going to come on the campus premises is showcased on the placement info page, addition, deletion and editing a company’s information is done by the administrator.
* Course Resources is a page where all the course content of subjects is displayed semester wise starting from semester 1 to semseter 8. A student can download the content directly from the website whereas the admin can upload and remove the files associated with a subject.
* Login Database is a database that contains the information of the users who are authorized to use the application. It is created on phpmyadmin platform and contians the login credentials of stuents,teachers and admin.
* Placement Information Database is a database also created on phpmyadmin platform and contians information regaring the placement drives that are currently happening in the college.
  1. **Scope of Future Enhancements**
* An attendence Sysytem
* A closed beta testing and get a detailed review on each and every aspect of the application.
* Update the application according to the reviews to get a user friendly result.
* To include ease of life features like E-Id card, student clubs information and enrollment, interactive campus map, timetable, performance, feedback and fees payment gateway.
* This application can be deployed to replace the existing college Erp system and can be used to manage students and Staff efficiently

**REFERENCES**

[1].https://code.visualstudio.com/learn/get-started/basics

[2]. https://www.jetbrains.com/pycharm/

[3]. https://getbootstrap.com/

[4]. https://www.apachefriends.org/

[5]. https://www.phpmyadmin.net/

[6]. https://www.javatpoint.com/

[7]. https://www.tutorialspoint.com/index.html

[8]. https://www.python.org/

[9]. https://html.com/

[10]. https://www.w3schools.com/html/

[11]. <https://flask.palletsprojects.com/en/2.2.x/quickstart/#a-minimal-application>

[12]. https://www.figma.com/files/recent?fuid=1088869950393308809