SUMMER TRAINING REPORT

ON

‘ERP FOR SCHOOL’

Submitted to

RAJASTHAN TECHNICAL UNIVERSITY

In Partial Fulfilment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER SCIENCE AND ENGINEERING

BY

Kaushal Saraswat (17ESKCS076)

UNDER THE GUIDANCE OF
Sushant Kumar

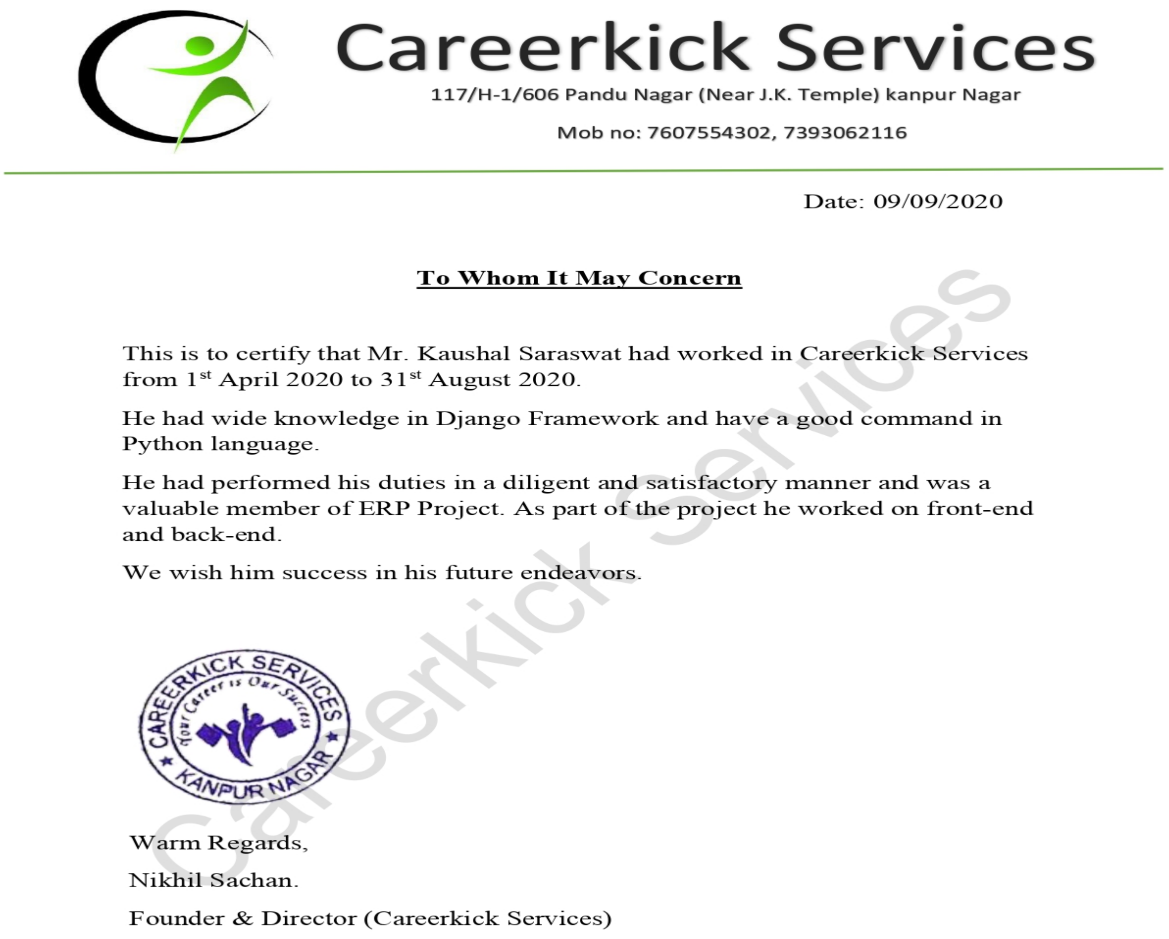
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SWAMI KESHVANAND INSTITUTE OF TECHNOLOGY ,
JAIPUR

2020-2021

Swami Keshvanand Institute of Technology, Jaipur

Department of Computer Science and Engineering

CERTIFICATE



Careerkick Services

117/H-1/606 Pandu Nagar (Near J.K. Temple) kanpur Nagar

Mob no: 7607554302, 7393062116

Date: 09/09/2020

To Whom It May Concern

This is to certify that Mr. Kaushal Saraswat had worked in Careerkick Services from 1st April 2020 to 31st August 2020.

He had wide knowledge in Django Framework and have a good command in Python language.

He had performed his duties in a diligent and satisfactory manner and was a valuable member of ERP Project. As part of the project he worked on front-end and back-end.

We wish him success in his future endeavors.

CAREERKICK SERVICES
Your Career is Our Success
KANPUR NAGAR

Warm Regards,

Nikhil Sachan.

Founder & Director (Careerkick Services)

Acknowledgement

It is my pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior, and acts during the course of study.

I express my sincere gratitude to Dr Mukesh Kumar Gupta, HOD, Computer Science for providing me an opportunity to undergo summer internship at CAREER KICK SERVICES I am thankful to Mr NIKHIL SACHAN for his support, cooperation, and motivation provided to me during the internship for constant inspiration, presence and blessings. I also extend my sincere appreciation to Mr. SUSHANT KUMAR who provided his valuable suggestions and precious time in accomplishing my report.

At last I must express my sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

KAUSHAL SARASWAT
17ESKCS076

Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Django

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

1.2 OVERVIEW OF DJANGO

Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, XML, etc). The site you are currently reading is built with Django!
Internally, while it provides choices for almost any functionality you might want (e.g. several popular databases, templating engines, etc.), it can also be extended to use other components if needed.

1.2.1 Secure

Django helps developers avoid many common security mistakes by providing a framework that has been engineered to ”do the right things” to protect the website automatically. For example, Django provides a secure way to manage user accounts and passwords, avoiding common mistakes like putting session information in cookies where it is vulnerable (instead cookies just contain a key, and the actual data is stored in the database) or directly storing passwords rather than a password hash.

1.2.2 Scalable

Django uses a component-based “shared-nothing” architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Having a clear separation between the different parts means that it can scale for increased traffic by adding hardware at any level: caching servers, database servers, or application servers. Some of the busiest sites have successfully scaled Django to meet their demands (e.g. Instagram and Disqus, to name just two).

1.2.3 Portable

Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavours of Linux, Windows, and Mac OS X. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

Chapter 2

DJANGO

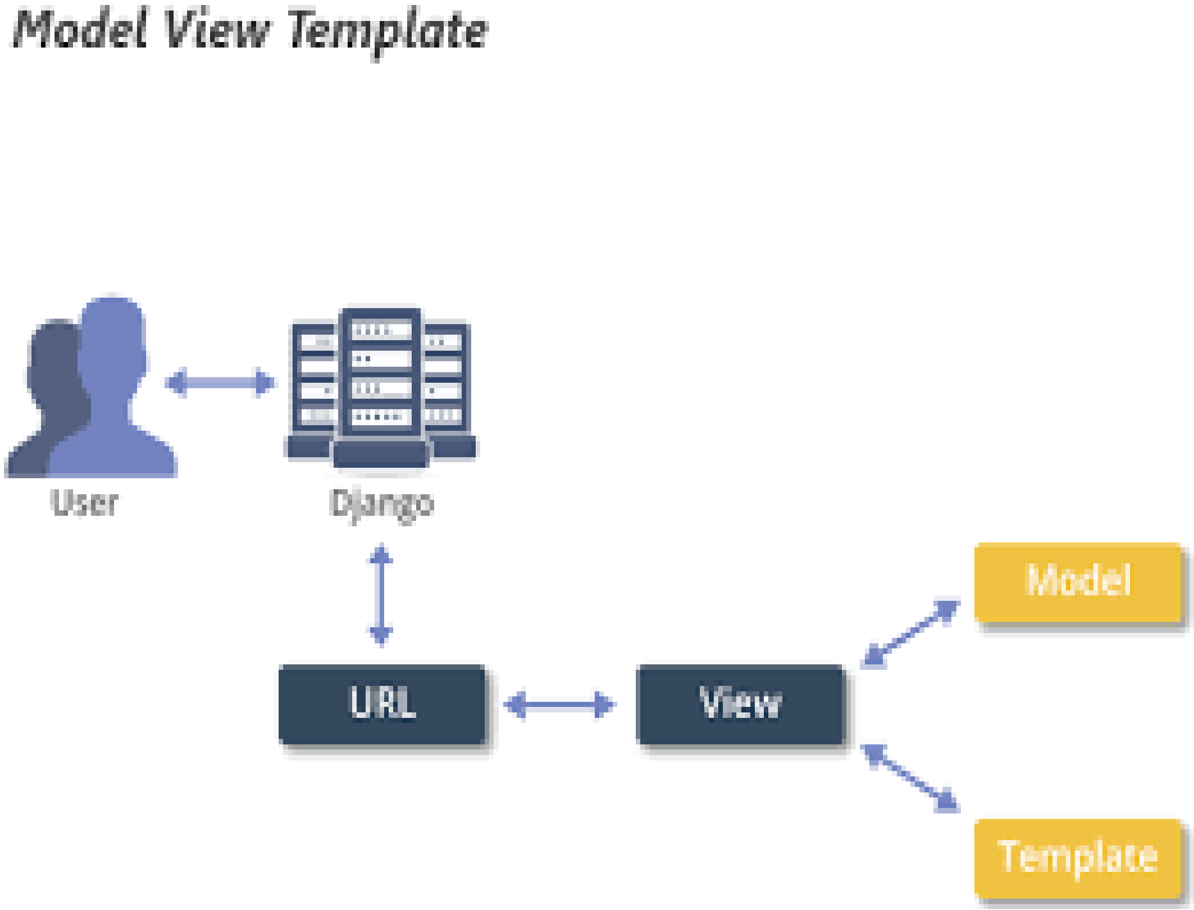


Figure 2.1: DJANGO ARCHITECTURE

2.1 ABOUT

The Template handles the UI and architecture part of an application. The view does the logical part of the application and interacts with the Model to get the data and in turn modifies the template accordingly. Here as already mentioned, Django works as a controller and gets a URL that is linked to the view part of the application and thus transports the user responses to the application. This complete interaction is dealt with this Django MVT architecture. When we create a project, there would some default files that would be created.

2.2 MODELS

A model is the single, definitive source of information about your data. It contains the essential fields and behaviors of the data you're storing. Generally, each model maps to a single database table.

2.3 VIEWS

View function, or view for short, is a Python function that takes a Web request and returns a Web response. This response can be the HTML contents of a Web page, or a redirect, or a 404 error, or an XML document, or an image . . . or anything, really. The view itself contains whatever arbitrary logic is necessary to return that response. This code can live anywhere you want, as long as it's on your Python path. There's no other requirement–no "magic," so to speak. For the sake of putting the code somewhere, the convention is to put views in a file called views.py, placed in your project or application directory.

2.4 TEMPLATES

Being a web framework, Django needs a convenient way to generate HTML dynamically. The most common approach relies on templates. A template contains the static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted. For a hands-on example of creating HTML pages with templates,

Chapter 3

FRONT-END USED

3.1 HTML5

Figure 3.1: HTML

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

3.2 CSS

Figure 3.2: CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

3.3 BOOTSTRAP

Figure 3.3: BOOTSTRAP

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components

3.4 JavaScript

Figure 3.4: JAVASCRIPT

JavaScript (JS) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles. Read more about JavaScript.

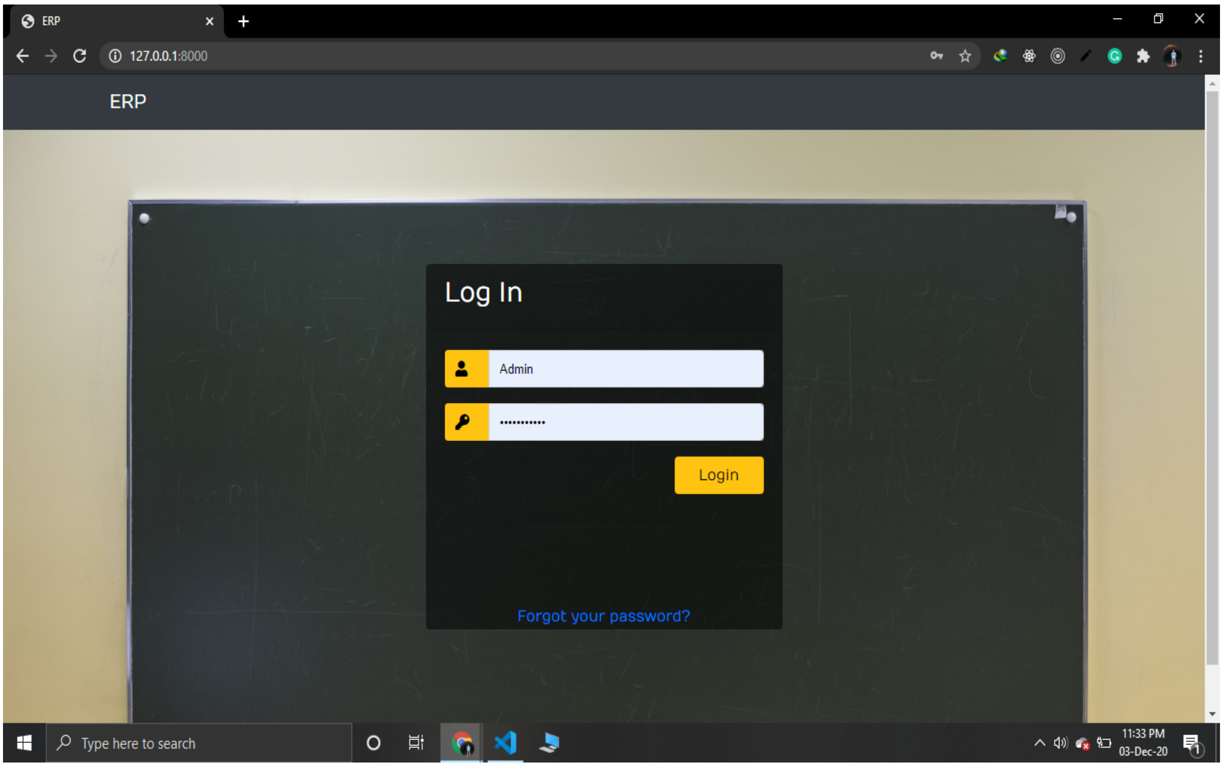
This section is dedicated to the JavaScript language itself, and not the parts that are specific to Web pages or other host environments. For information about API specifics to Web pages, please see Web APIs and DOM.

Chapter 4

ERP FOR SCHOOL

4.1 ERP LOGIN

Login Page for the ERP..!!



Admin

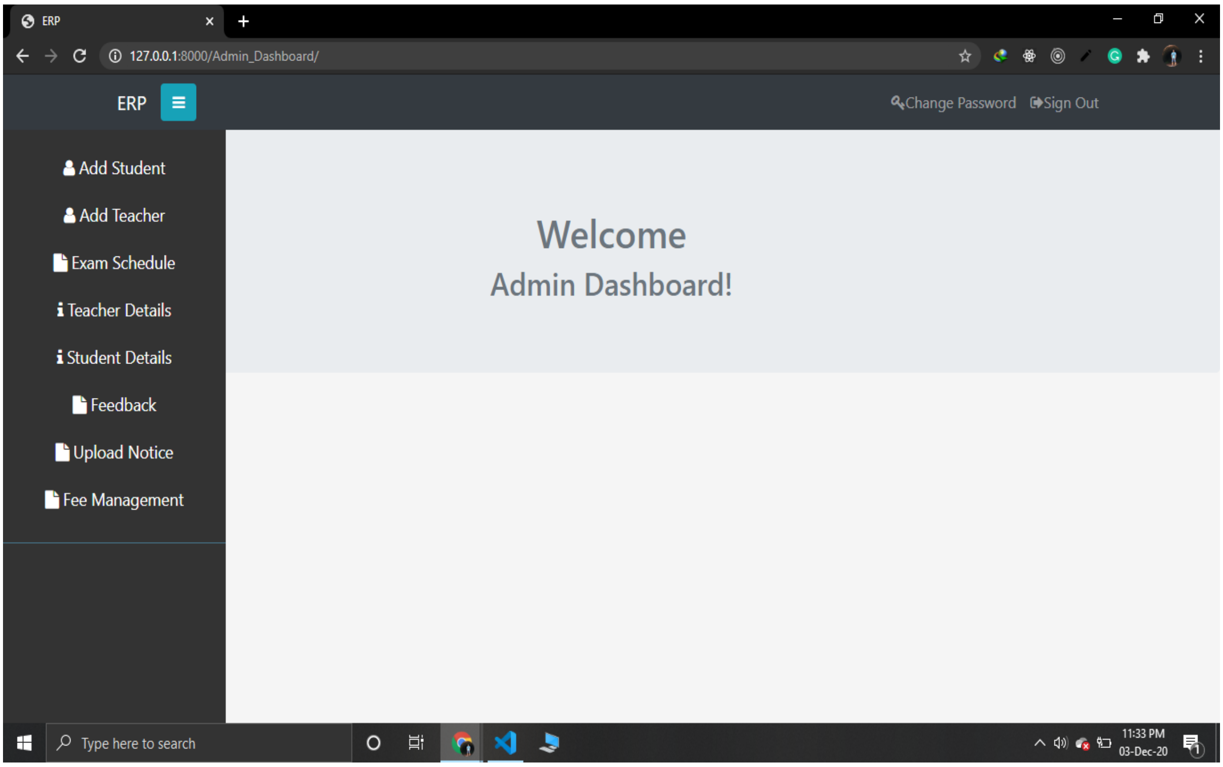
•••••••••••

Figure 4.1: LOGIN PAGE

4.2 ADMIN MODULE

Admin has following functionalities:

- Add Student
- Add Teacher
- Exam Schedule
- Teacher Details
- Student Details
- Upload Notice



Welcome

Admin Dashboard!

Figure 4.2: ADMIN DASHBOARD

4.3 TEACHER MODULE

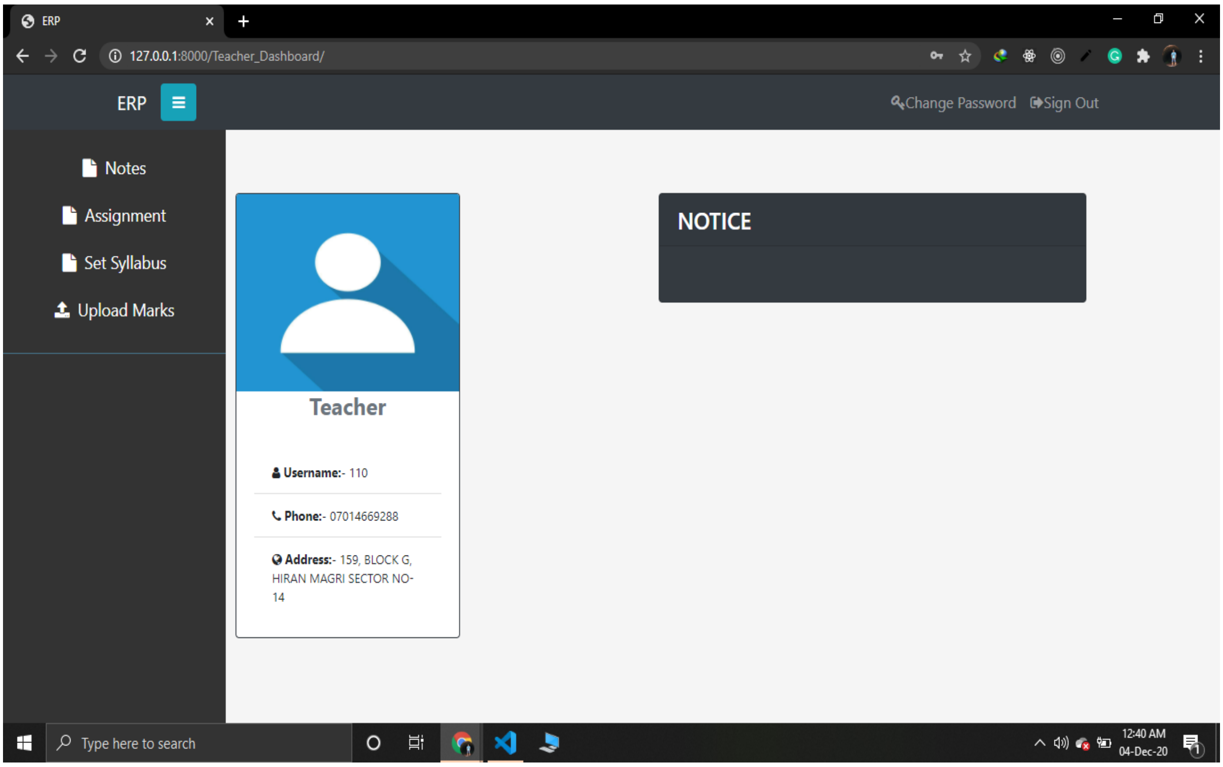
Teacher can be two types:

- Subject Teacher
- Class Teacher

4.3.1 Subject Teacher

Subject Teacher has following functionalities:

- Upload Notes
- Upload Assignment
- Set Syllabus
- Upload Marks



127.0.0.1:8000/Teacher_Dashboard/

Username:- 110

Phone:- 07014669288

Address:- 159, BLOCK G, HIRAN MAGRI SECTOR NO-14

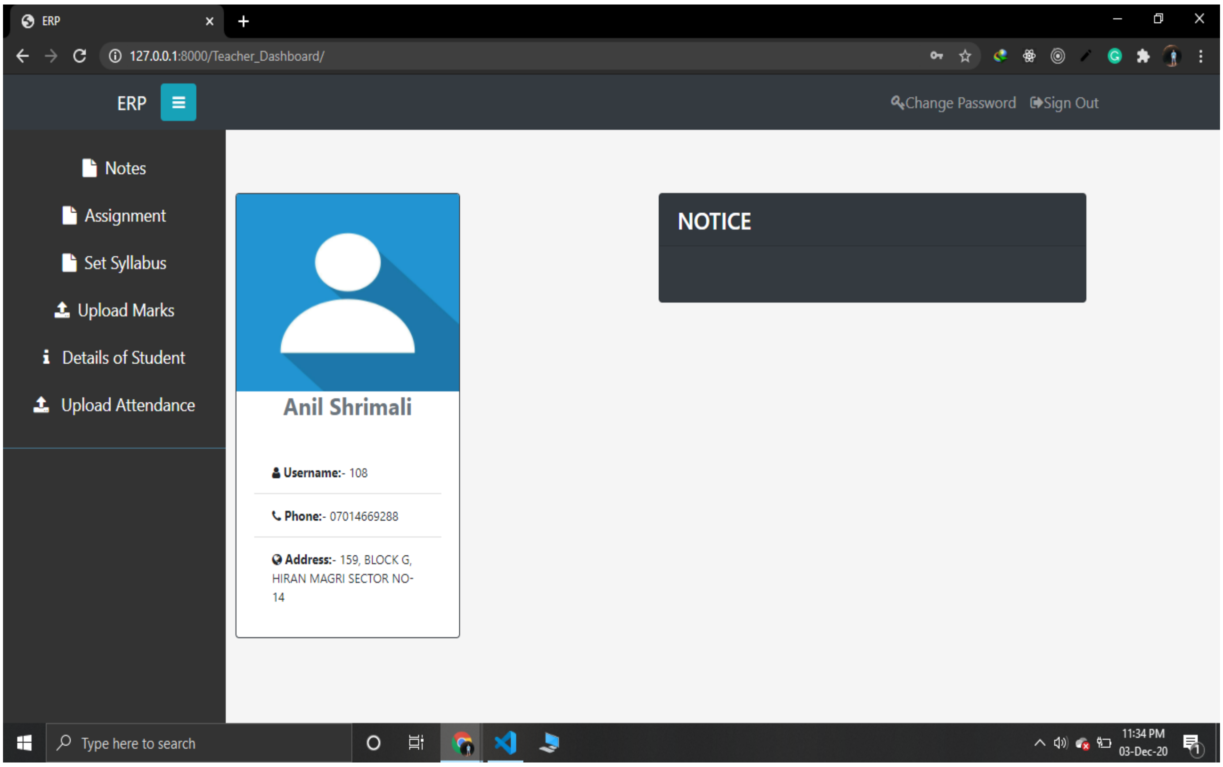
Figure 4.3: Subject Teacher Dashboard

4.3.2 Class Teacher

Class Teacher has only one additional functionality that is

- Upload Attendance

.



Anil Shrimali

Username:- 108

Phone:- 07014669288

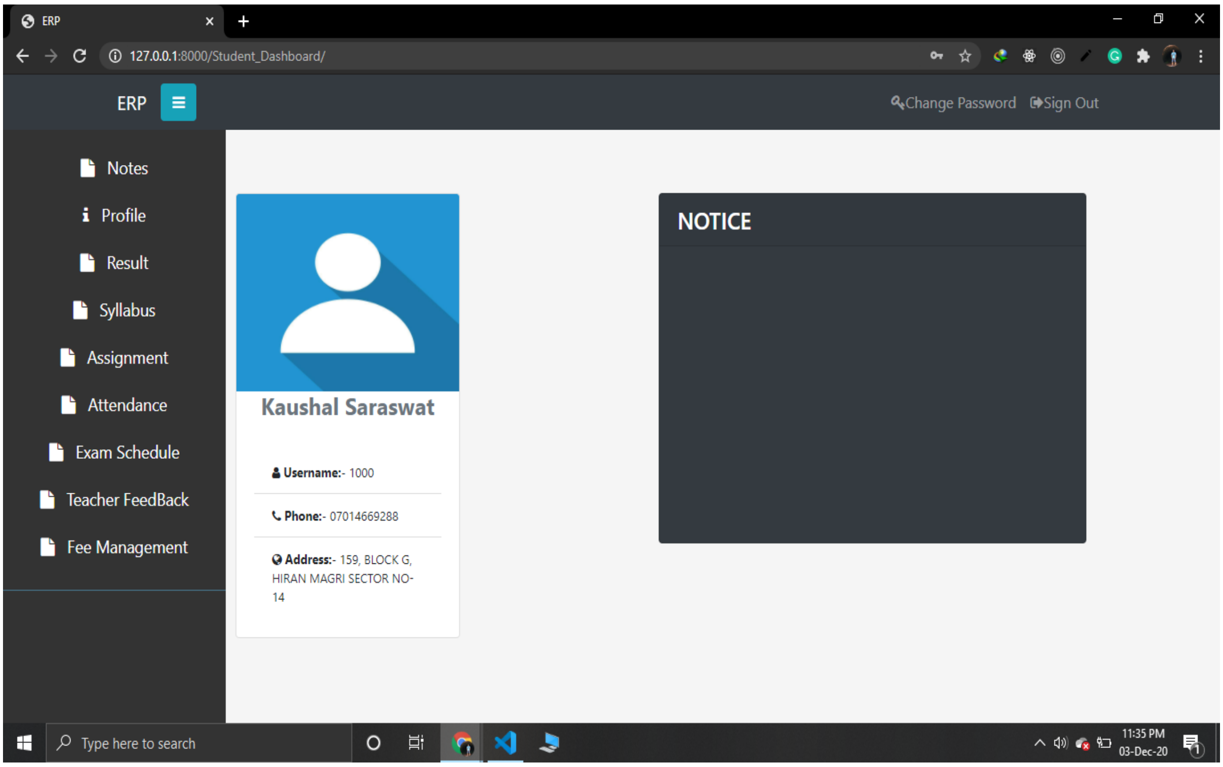
Address:- 159, BLOCK G, HIRAN MAGRI SECTOR NO-14

Figure 4.4: Subject Teacher Dashboard

4.4 STUDENT MODULE

Student has following functionalities:

- Download Notes
- Submit Assignments
- View Results
- View Attendance
- View Exam Schedule
- Submit Teacher Feedback



Kaushal Saraswat

Username:- 1000

Phone:- 07014669288

Address:- 159, BLOCK G, HIRAN MAGRI SECTOR NO-14

Figure 4.5: Student Dashboard

Chapter 5

CONCLUSION

5.1 TAKEAWAYS OF TRAINING

In my complete training period I have worked as a web developer (full stack) and learned and experience the key features of django framework and front-end part
In this training i also learned about how to work together as a team and delivery the assign task on time so the balance of the team is not disturbed

5.2 FUTURE SCOPE

Django is a tool that has all the packages required for web development, and the future of Django is very safe and bright. You can understand this by checking out the below points.

- Django has more number of generators when compared with other web development languages, management tools for dependency, different libraries, API support.
- Django has an integrated CSS framework.
- Django is a Python-based modern tool that is used extensively on the back end of websites.
- Django has numerous client-side dependencies.
- Django is a client-side Python framework similar to Angular.js or other JavaScript-based frameworks.