Project Requirements

1. PYTHON

We are using Python as a core compact to develop our website, as it is very versatile with various types of operation in the world of computer science. It is a widely used general-purpose, high-level programming language and with the support of rich libraries, it becomes the best choice for us to use Python.

1. IDE – Pycharm

We are Pycharm as our IDE because its suites are convenient and easy to maintain directories and manage various commands on the terminal.

1. DJANGO

Django is a back-end server-side web framework that makes it easier to build web pages using Python. It takes care of all the major stuff in order to build the website more effectively and efficiently.

We choose to go with it because it emphasizes the reusability of the components, also referred to as DRY(Don’t Repeat Yourself), and it comes with ready-to-use features like a login system, database connection, and CRUD operations.

As we are building an Applicant Tracking System(ATS), in which the database plays a major role, Django can be very handy for this purpose.

Django follows the MVT design pattern-Model View Template, Model-The data you want to present, usually from a database, View-A request handler that returns the relevant template and content – based on the request from the user, and Template-A text file containing the layout of the web page, with logic on how to display the data.

Model

In Django, the data is delivered as an ORM-Object Relational Mapping, which is a technique designed to make it easier to work with databases. The models are usually located in a file called maodels.py.

View

A view is a function or method that takes HTTP request as arguments, imports the relevant models, find out what data to send to the template, and returns the final result.

The view is usually located in a file called views.py.

Template

A description of how the result should be displayed.

We are using HTML to describe our code and the layout of a webpage. Templates are normally present in the directory called templates.

URLs

Django also provides a way to navigate around the different pages of a website. Whenever the user requests a URL, Django decides which view it will send it to.

This is done by a file called urls.py.

Complete Procedure:

1. Django receives the URL, checks the urls.py file, and calls the view that matches the URl.
2. The view described in the views.py, checks for relevant models.
3. The models are imported from the model.py file.
4. The view then sends the data to a specified template in the template folder.
5. The template contains the HTML tags and the data it returns finished HTML content back to the browser.
6. HTML

We are using HTML, Hypertext Markup Language for creating Web Pages.

It describes the structure of the web pages and contains a series of elements and these elements tell the browser how to display the content.

We have used version HTML 5.

1. CSS

With the combination of HTML and CSS we have designed and styled our web pages. CSS(Cascading Style Sheets) is the language we used to style out HTML document and it describes hot HTML elements are to be displayed on screen, paper or in other media. It saves a lot of words, it can control the layout of multiple web pages all at once.

We have used internal CSS in our project.