For better understanding and interactivity some of this document‘s idea is thought to be implemented as a web application.

Therefore, from the many available programming languages and frameworks for web applications HTML, CSS and, JAVASCRIPT for frontend and DJANGO (PYTHON’s FRAMEWORK) for back-end development were chosen.

Throughout the development process the languages chosen were HTML, CSS3, JAVASCRPIT 1.8.5, PYTHON3, DJANGO 3.0.3 and the application used to write the codes on or IDE was VS CODE 1.5 as of that time these were the latest versions of the applications. Some other additional plugins and extensions were also used in few places needed.

But first here is some overview with the technical words:

Web application: A web application or web app, short for web-based application, is software that runs in an [Internet browser](https://www.computerhope.com/jargon/b/browser.htm). A web application provides a user interface, offers utility or entertainment, and the ability to access, create, store, or modify data.

Frameworks: A Framework is a platform used as a foundation for developing applications. It provides a structure in which software developers can build programs for a specific platform. For example, a framework may include predefined classes and functions that can be used to process input, manage hardware devices, and interact with system software.

Front-end: Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The objective of designing a site is to ensure that when the users open up the site they see the information in a format that is easy to read and relevant.

Back-end: The backend usually consists of three parts: a server, an application, and a database. Backend technologies usually consist of languages like [PHP](http://teamtreehouse.com/library/topic:php), [Ruby](http://teamtreehouse.com/library/topic:ruby), [Python](http://teamtreehouse.com/library/topic:python), etc. To make them even easier to use they’re usually enhanced by frameworks like Django.

Now there rises a question “Why were all these features chosen for this project?“.

***Why build a web application?***

The reason for the software to be a web app was because in most cases, web apps need less RAM to run and also unlike desktop apps they don’t need local installation and can be accessed through the browser once the server is set. Thus, to start building the web app HTML, CSS and JAVASCRIPT were needed. These languages are used for the front-end development.

**HTML**provides the basic structure of sites, which is enhanced and modified by other technologies like CSS and JavaScript.

**CSS** is used to control presentation, formatting, and layout.

**JavaScript** is used to control the behavior of different elements.

***Why use Python?***

Python is a high-level programming language and the basis of many web frameworks. Since it has easy syntax, programmers are able to complete coding in fewer steps compared to Java or C++. Unlike other languages, Python was engineered as a full stack programming language regardless of web development. Thanks to its libraries and support, Python is an excellent option for any project, mobile app, web app, IoT, data science or AI. The programmers are easily able to tell where everything originates and leads, making it relatively simple to learn and easy to debug. Furthermore, numerous companies use Python for their project development. For example, the famous social network, ***Instagram*** uses Python 3. Another example is ***Amazon* and*NASA*** also uses Python.

***Why use Django?***

Django is an open-source framework for backend web applications based on Python-one of the top web development languages. Its main goals are simplicity, flexibility, reliability, and scalability. Django has its own naming system for all functions and components (e.g., HTTP responses are called “views”). It also has an admin panel, which is deemed easier to work with than in Lavarel or Yii, and other technical features, including:

Simple syntax, It’s own web server, MVC (Model-View-Controller) core architecture, a Python unit test framework.

Django is fast, simple, secure and can withhold multiple requests at a time. Because of it’s features django is being used by many large companies like Instagram, Spotify, The Washington post, Mozilla, Pinterest and others.

The above reasons were the majors ones why those languages and frameworks used.

The System

The web application is an implementation of what the Ethiopian Road Authority Project Management would be like. The main idea is to facilitate the internal workings of the company by paving a way for its project management system, currently done in papers and pens, to be digitalized.

Since, the ERA is a huge company with many employees and departments and/or directorates it is way more broad to build a web application compatible for the entire company. Therefore, this Demo system is done for one a selected Directorate.

When done now it’s planned and assumed that a directorate will have :

A Director – a person who is in charge of the directorate.

An Office Assistant – a person assistant to the office and the director.

Record officer – a person who manages and records the data, files, and letters entering and leaving the office.

Human Resources Officer – a person in charge of the controlling the resources of the office and also it’s employees.

Team Leader – a person leading a team in the directorate answers to the director.

Secretary – a person who is an assistant to the team leader and the team.

Lead Engineer – a person under a team and answering to the team leader.

Project Engineer – a person who is assigned a project to answers to the lead engineer.

An Admin – a person who controls the servers and manages the IT infrastructure.

With all of these actors and each of them having their own