**Project Information**

Local server address: localhost:8000 or http://127.0.0.1:8000/

Admin username: admin

Admin password: admin

Terminal Commands

* python manage.py runserver – – Run the locally hosted development server
* python manage.py makemigrations – – Create the migration files in order to create tables within the database
* python manage.py migrate – – Create the actual tables within the database according to the migration files generated by the makemigrations command
* python manage.py createsuperuser – – Create a super user with all admin privileges
* python manage.py shell – – Give the terminal access to run sql commands and check database connections and content

Miscellaneous Information

* localhost:8000/admin – – Bring up the django administration interface. All registered models can be viewed, and interacted with directly through the django admin web interface.
* Activate your virtual environment while cd’d within the project directory (if you are using it with the project): env\scripts\activate

EzBooks\_Site Information

* This is the location of all global settings within the project. No actual project content/code goes within this directory, this directory just points the framework to where the content/code for the project is located.
* Connection setting for the database are located within the settings.py file within this directory
* All project apps must be registered in the INSTALLED\_APPS array within the settings.py file in this directory.
* All project apps must have urls registered in the urls.py file.

Users Information

* This directory is an app within the whole of the project.
* The User app controls the creation, and manipulation of user profiles and allows for the connection of different data to the users profile.
* This app also controls the authentication of logging in users with their appropriate passwords.
* The models within the models.py file of this app control the storage of the user profiles within the database
* The forms within the forms.py file of this app allow data to be entered into the models through the web interface.
* Paths to the different web pages of the users app are all registered within the urls.py file.
* Views within the views.py file contain the information needed to render the appropriate web page.
* Templates are contained within a nested directory, according to framework requirements. The templates are the actual html for the webpages. (templates, html, and css are discussed later)
* The main reason for the creation of this app was to separate the user creation and management from the main section of the project for both security and simplicity.

ez\_main Information

* This directory is an app within the whole of the project.
* ez\_main is the location of the main code for the functionality of the project and is considered the central hub of the project itself.
* Models within this app are for the main storage of needed project data. This includes the books, the classes, and specific pieces of information which are associated to different users (the users class schedule).
* Base.html is a template located within this directory which is used globally for all other webpages in the project, including webpages which are contained within seperate apps
* All scripts and stylesheets must be linked within the base.html since this template is the only template to contain an html head element. (This is due to the inheritance feature being used on the other webpages).
* The ez\_main directory is the place for the main code for the functionality of the project unlike that of the users directory which is used exclusively for the creation and management of the user profiles.

Web pages and UI Information

* Each of the web pages for the project are rendered from templates which are stored within each of their appropriate apps contained within the project.
* Each template is sent the needed data, and are rendered through a view, contained in the same app as the template.
* Templates are portions of html, since the project takes advantage of the ability to inherit from other templates in order to minimize the amount of repeated html code.
* All sub-templates of the project inherit from the base.html file contained within the ez\_main app.
* Base.html contains the html head information, which in then inherited by all other web pages.
* Base.html also contains html for the top navbar, and footer which are global to all the web pages of the project.
* Stylesheets/images/javascript used for templates should be stored within the static directory contained within the same app as the template which makes use of the static file.
* When linking stylesheets/images/javascript django requires {% load static %} to be specified before the link is made, but only needs to be specified once per template. Loading static does not get inherited along with the base.html.