**System Documentation**

**Group 13**

Scheduling Project

**Team Members**

Anisha

Daniel

Nawaf

Stanley

# Contents

1 Introduction 4

2 Setup Network

3 Setup Database

4 System Maintenance

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Name** | **Description** |
| 1 | 4/22/2020 | Scheduling Project | Track the class ID, class time, Professor availability and classrooms within SCE department |
|  |  |  |  |

# Introduction

The Scheduling Project is a web application that will be accessible to any computer that is connected to the UMKC network. The Scheduling Application helps assist schedule creators in their design of semester schedules by making it easier to create and alter schedules without breaking constraints set by professors. The web application is a Django project developed with Python 3.8 and uses a SQLite database.

# Setup Network

The Scheduling Project web application must be deployed to UMKC’s network and available to the schedule creators. This can be done through the UM system or through a Unix/Linux server depending on resources and the amount of times that this website may be used concurrently. For further deployment questions, here is Django’s documentation on how to deploy a project: <https://docs.djangoproject.com/en/3.0/howto/deployment/>.

# Setup Database

Since our project utilizes SQLite, a third-party database to setup and maintain a database such as SQLserver or PostgreSQL are not needed. For our system to function correctly, the database must be populated with all the professors, classes, rooms, times and constraints so that our system is able to correctly create a schedule. Failure to add one of the above items may cause issues with scheduling as certain constraints may be violated without an easy solution to fix.

**Populating Database**

1. To populate the database, navigate to the database tab.
2. Click on the drop-down table of the item you wish to add or edit. (Professor, class, room etc.)
3. Once the drop-down table is expanded, click on the plus arrow undeath the title of the table which will navigate you to the object creation screen.
4. Enter the objects information and click save.

Although this seems like a long-winded approach to enter massive amounts of information, we found this to be the easiest way and most user-friendly way of entering information. Integration with services that already contain this information is a feature to be added in the future. Note that this is one time set up and will only needed to be changed when objects are either added or deleted.

# System Maintenance

Our system is written in Python 3.8 utilizing the Django framework. The user interface contains three main sections, a log-in screen and an object creation screen. Each tab has its own html sheet which is based off base.html which contains objects present in every tab, such as the navigation bar. Each tabs html file name is named after the tabs name making it easy to locate.

When the application is first ran the application displays the log-in screen where only validated schedule creators can log in. Once that user is logged in and verified, the main page is displayed showing a video tutorial on how our system functions and a navigation bar that links to the tabs “Home”, “Database”, and “Generate”.

Our system follows the model view controller philosophy and the logic for classes and the database models appear in the models.py file. This file shows all the models and altering these affects that object in the database. Direct access to the SQL database is available to super users through the admin page.