Table of Contents

[1 Set Up 2](#_Toc69806065)

[1.1 Prepare Documents 2](#_Toc69806066)

[1.2 Launch Web App 2](#_Toc69806067)

[2 Navigate Web App 3](#_Toc69806068)

[2.1 Home Tab: Generate New Schedule 3](#_Toc69806069)

[2.1 Home Tab: Display Past Schedule 4](#_Toc69806070)

[2.2 Timetable Tab 5](#_Toc69806071)

[2.3 Summary Tab 6](#_Toc69806072)

[2.4 ICU Duties Tab 7](#_Toc69806073)

[2.5 Points Tab 8](#_Toc69806074)

[3 Code Documentation 9](#_Toc69806075)

# 1 Set Up

## 1.1 Prepare Documents

1. Put *“flask\_app”* folder in C-drive.
2. Fill up the information (Roster, Duties, Training, Priority Leave, Public Holiday) in “*information\_excel.xlsx”* accordingly.

***\*Please input the dates in M/DD/YYYY format eg. 2 August 2020 should be inputted as 8/02/2020.***

***\*Please do not rename the sheets.***

1. Open *“Outlook”* application and rename the folder which receives FormSG submission to “*test”* (without the quotation marks).

\*For testing purposes, import backup.pst into Outlook and rename the folder as “test” for examples of FormSG submission.

## 1.2 Launch Web App

1. Open the “*command prompt*”, then copy & paste the following:

cd C:\flask\_app

env\Scripts\activate

set flask\_debug=1

flask run

Text

Description automatically generated

This means that the application is running. If you see an error that says python not found,

refer to [Appendix A](#_Appendix_A:_Python).

1. Open a web browser (Google Chrome) and go to <http://127.0.0.1:5000/>

# 2 Navigate Web App

A screenshot of a video game

Description automatically generated with medium confidence

## 2.1 Home Tab: Generate New Schedule

1. Click on the *“New Schedule”* button to generate a schedule and fill up the necessary details.

***\*For testing, fill up with “August”, “01-08-2020”, “15-08-2020”.***

1. Click on the *“Edit Constraints”* button if you wish to change the constraints used for LP.
2. Click *“Generate”* button and you will be taken to *“Timetable”* tab if all constraints are met.

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

1. If there is a clash in *“information\_excel.xlsx”*, you will be redirected to the following page instead. If this happens, fix the excel accordingly.

Table

Description automatically generated

1. Go back to the *“home”* tab and re-do step 1-3.

## 2.1 Home Tab: Display Past Schedule

1. Put the CSV of the past schedules (*“scheduleJ.xlsx”* and *“scheduleS.xlsx”*) in *“flask-app”* folder. Refer to section 2.2 step 6 on how to generate the CSV.
2. Click *“Generate”* button and you will be taken to *“Past Timetable”* tab.

## 2.2 Timetable Tab

A picture containing calendar

Description automatically generated

1. Hover over the word in the cell to see the comments from FormSG as a tooltip.
2. Click on the 3-dots to change the remarks. Comments are automatically updated in the tooltip. Otherwise, refresh the page.
3. Click on the word in the cell to change the word eg. Working to Duty. The changes are automatically updated in the cell. Otherwise, refresh the page.
4. Table

   Description automatically generatedGraphical user interface, text, application, chat or text message

   Description automatically generatedGraphical user interface, text, application, chat or text message

   Description automatically generatedAfter making all the changes in step 3, click on the *“Validate”* button to check if the changes fulfil the constraints. A pop-up will appear to point out the unmet constraints, if any.
5. Click on the *“Download as PDF”* button to download the schedules as PDF. Check *“flask\_app”* folder for the PDF (*“timetable\_junior.pdf”* and *“timetable\_senior.pdf”*).
6. Click on the *“Export as CSV”* button to download the schedules as CSV, if you want to display it in the web app in the future. Check *“flask-app”* folder for the CSV (*“scheduleJ.xlsx”* and *“scheduleS.xlsx”*).

## A picture containing graphical user interface Description automatically generated2.3 Summary Tab

1. Click on the *“Download as PDF”* button to download the call summary as PDF. Check *“flask-app”* folder for the PDF (*“summary.pdf”*).

## 2.4 ICU Duties Tab

Graphical user interface, text

Description automatically generated

1. Fill up the *“First Doctor”, “Second Doctor”, “Third Doctor”, “Fourth Doctor”* textboxes with the names of the 4 doctors who will be rotated in the ICU for the month, then click on the *“Confirm Doctors”* button.

*\*If any of the names are changed after clicking on the “Confirm Doctors” button, please refresh the page before proceeding.*

1. Click on the dropdown list to assign one of the 4 doctors to the date.
2. Click on the *“Download as PDF”* button to download the ICU duties as PDF. Check *“flask-app”* folder for the PDF (*“icu\_1.pdf”* and *“icu\_2.pdf”*).

## 2.5 Points Tab

A picture containing calendar

Description automatically generated

1. Click on the *“Download as PDF”* button to download the points summary as PDF. Check *“flask-app”* folder for the PDF (*“points\_junior.pdf”* and *“points\_senior.pdf”*).

# 3 Code Documentation

Graphical user interface, text, application

Description automatically generated

1. Navigate to *“C:\flask\_app\docs\\_build\html”* and double-click on *“index.html”.* This page will be displayed in the browser.
2. Click on *“Main Application (app.py)”* for documentation functions related to the webpages.
3. Click on *“Helper Functions (helperFunctions.py)”* for documentation on functions used in app.py.
4. Click on *“Linear Programming (lpFunction.py)”* for documentation on the LP code.