

EMERGENCY MEDICINE JUNIOR DOCTOR ROSTERING USING CONSTRAINT PROGRAMMING

PROJECT DESCRIPTION

The Singapore General Hospital (SGH) Emergency Department (ED) creates their junior rosters manually, having to meet doctor's shift requests, training requirements as well as daily staffing requirement. This process is tedious and extremely time consuming, taking the roster planner weeks to finalise the next month's roster.

The purpose of this project is thus to create an application that automatically generates a roster based on given requirements. This roster should have two views, namely a shift by day view or a doctor by day view.

	DAY 1	DAY 2	DAY 3
SHIFT A	Dr. A Dr. D	Dr. C	Dr. B
SHIFT B	Dr. B	Dr. A Dr. D	Dr. C
SHIFT C	Dr. C	Dr. B	Dr. A Dr. D

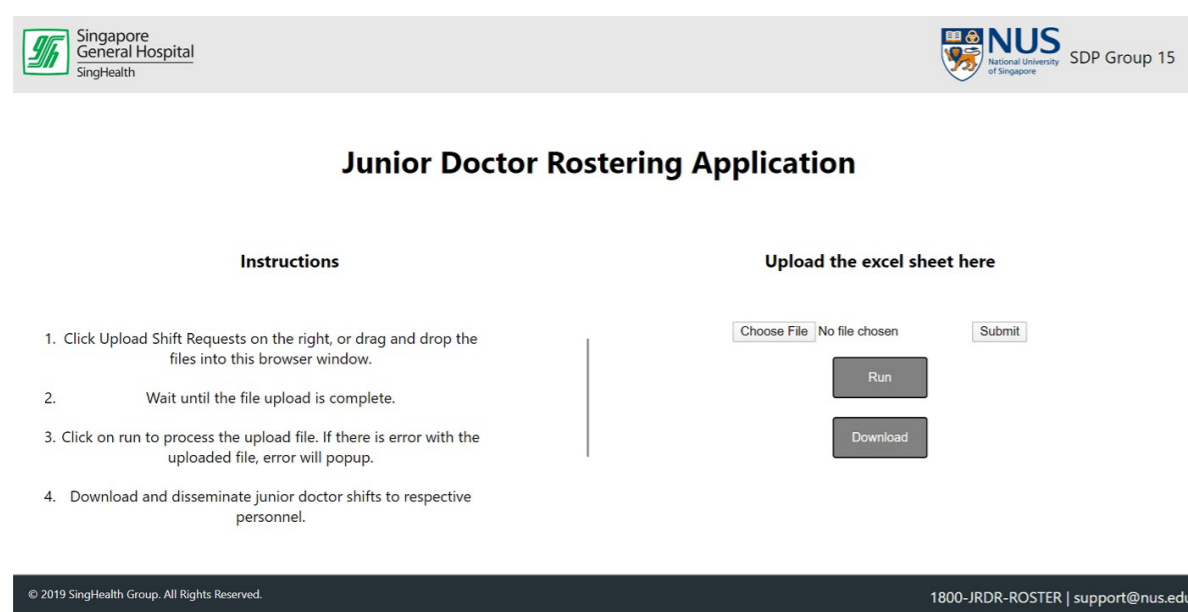
Shift by day view

	DAY 1	DAY 2	DAY 3
Dr. A	SHIFT A	SHIFT B	SHIFT C
Dr. B	SHIFT B	SHIFT C	SHIFT A
Dr. C	SHIFT C	SHIFT A	SHIFT B
Dr. D	SHIFT A	SHIFT B	SHIFT C

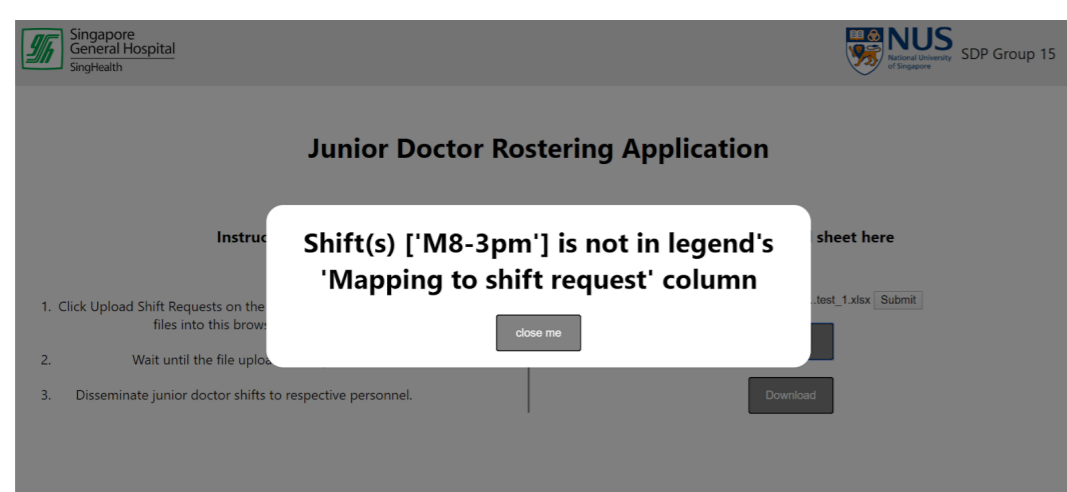
Doctor by day view

METHODOLOGY

We developed a web application with a simple User Interface (UI):



Through this UI, which was built using the React.js framework, the user can upload an excel file that contains the staffing requirements, shift requests and doctor information amongst other requirements. This information would be passed to the backend using Flask APIs and processed. Any errors with the input file would be displayed on the UI:



The model is then built, leveraging on the CP-SAT solver from Google OR-Tools. Amongst other model variables, a Boolean shifts variable is created which takes on the value 1 if doctor n is allocated shift s on day d and 0 otherwise:

$$shifts(n, d, s) = \begin{cases} 1, & \text{if shift } s \text{ is assigned to doctor } n \text{ on day } d \\ 0, & \text{otherwise} \end{cases}$$

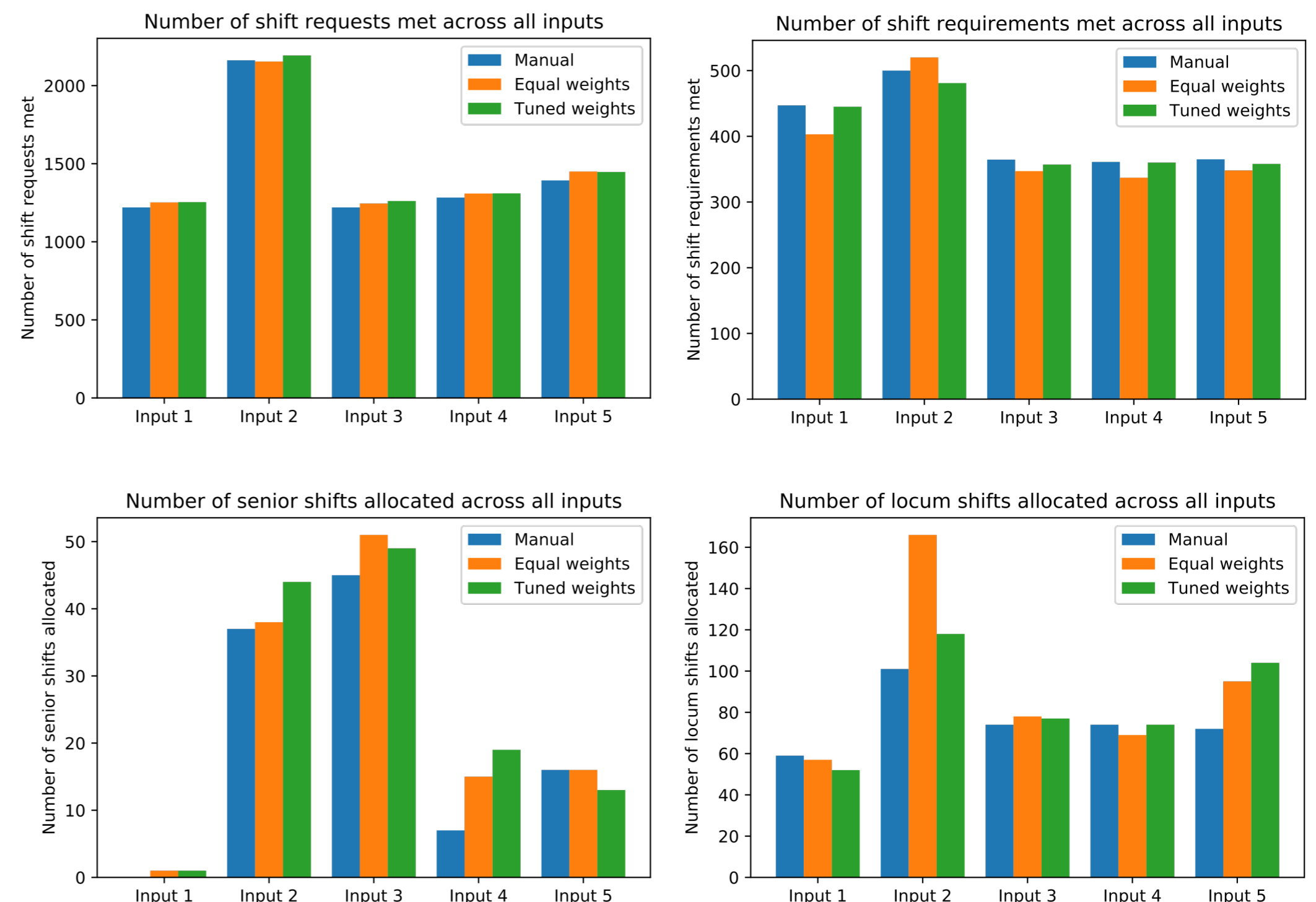
This model is created to be highly flexible to changes in the excel file such as addition of new doctors, changes in shift timings, etc. After which, constraints like "each junior doctor works only one shift a day" are added to the solver:

$$\sum_{s \in S} shifts(n, d, s) = 1, \quad \forall n \in \{\text{Junior doctors}\}, \quad d \in D$$

Under constraint programming, a generated roster that satisfies the above constraints are termed feasible rosters. But they may not be optimal. To optimise the roster, several objectives are defined whose weights could be altered using the excel file. Objectives include maximising sum of shift requests met and sum of shift requirements met while minimising number of senior and locum shifts allocated. An optimal roster is then generated and downloads can be made via the UI.

RESULTS & DISCUSSION

Rosters generated from the model is compared with five manually created rosters from different months:



As seen in the above figures, the rosters generated from the model outperforms those manually created mostly on the criteria of meeting doctors' shift requests amongst the comparison measures. This could be due to the ratio of the weights of the objectives described in the methodologies section.

Notably, the model with equal weights generated rosters with a lower number of shift requirements met as compared to those manually created. However, this can be rectified by tuning the weights for this objective such that more shift requirements would be met in rosters generated by this model.

The biggest improvement in a roster generated by the model versus those manually generated would be the time taken to create the roster. While the model takes minutes to hours to generate a roster, those manually created take up to two weeks. Furthermore, a time limit can be set for the model to be ran such that although an optimal solution is not reached, a feasible one with relatively good performance can be achieved.

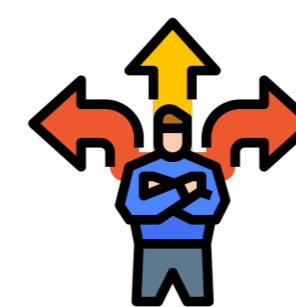
CONCLUSION

In conclusion, to improve the process of roster planning in SGH's ED, we developed a web application that is:



USER FRIENDLY

- Gentle learning curve
- Non-technical



FLEXIBLE

- Allows for multiple changes in the input
- Sustainable in the long term



ACCURATE & EFFICIENT

- Good performance
- Generated in minutes