# IB Fast Start Rotations

# FAT1: Sandeep Singh August 24, 2023

**I. Purpose/Objective**

The purpose of this document is to highlight and discuss my project regarding inbound role rotation and faster start up time. The goal of the project is to make the inbound process more efficient and ultimately save the site time, money, and resources in the process. My project consists of python code along with a couple excel sheets designed to rotate people around and let them scan their badge to get their role similar to what the Staffing Command Center (SCC) does.

**II. Background**

Prior to my project, the inbound department lacked a proper rotation as well as a fast and efficient way to start up shift. Roles would be handed out to associated after stand up. Associates would form a line and be given a task to do on the spot. This allows for human favoritism and an unproper role rotation. My project removed all of these. The program auto generates with randomization to allow for proper rotation and no favoritism to be allow.

The project includes a python code file that reads several excel sheets including data such as logins, their trained roles, who is clocked in, and what roles are needed for the day and how many people for that role are needed **(Appendix A).** The following are considered to automate for the roles/associates for the day. It has many features to make sure everything is randomized given the head count that has clocked in, such as pythons built in randomization feature.

Once the python code is finished running, the information such as roles and logins can be copied on the “scanforrole” excel sheet. This is where the associate will scan their badge and their corresponding role will be shown on the screen **(Appendix B).** There are behind the scenes features for managers to look at once all the scanning is done. The associates who have scanned will be highlighted red as well put into a list so at the end managers know which associates have yet to scan and what positions need to be filled still.

The program all together has many features to make the usage feel clean and straightforward. This coupled with detailed step by step instructions should make it very easy for a new user to understand and handle the role rotation program.

**III. Experimentation Process**

When it comes to any coding project there are always bugs that need to get addressed. I made sure to test the code in as many ways possible to address any bugs I could find before rolling it out. The plan was to fix whatever bugs come before and after rolling out. After the first launch, we ran into a couple of issues that we had to address and after discussion with the team, we were able to fix them for the next day. This is how code is built on and improved and is slowly perfected. The feedback helped immensely and after a bit of more testing the project kept getting better and running more smoothly.

**IV. Experimentation/Data**

After the first two runs of the program, we were able to significantly improve the code and strategy. We were able to bring the startup time down to as low as 7 mins on certain days. Using the FastStart website, it was evident that FastStart times vary between shift to shift. This maybe due to many factors such as head count, volume, and staffing. The program is designed to tackle these solutions. After running, the average time of Start of Shift (SOS) is under the target of 10 mins (8m 31s) after 5 test cases **(Appendix C).** As well as bring the Total SOS Time lost down to an average of 12m 01s. However, this is not perfect, it can be improved on as the program is used.

The program can handle high head count where inbound must open multiple lines for decant and multiple trailers. Doing this manually, such as prime week leads to higher time lost.

* Prime week AVG SOS TIME DAY: 11 minutes **(Appendix D)**

The program can target not only decant, but also the dock as well which isn’t calculated by FastStart.

With lower head count the result tend to deviate slightly as staffing can be done just as fast when done manually, but what this doesn’t consider is the rotation. Many people can be seen doing the same task each day. Often, this leads to people complaining about how there is favoritism, but the program being completely random, removes that factor. It also makes sure that everyone decants at some point. This makes sure decant permissions are not lost which in the end costs the facility time and recourses to retrain the associates.

**V. Risk/Blockers**

As with all technology, there is always a risk of it not working. This is also the case for the program as well. The program uses the FCLM website to get a list of all the clocked in associates to give those associates the roles. The problem is the FCLM website is a piece of technology that can sometimes not work. In this case, manual roles will have to be assigned.

Another lowlight is the precedent among associates. Associates are used to getting the small role each day. This was seen on the first few runs of the program. Associates would question the role they got which rose the FastStart time. However, after the first few runs, associates started questioning it less and the process became a lot smoother.

**VI. Forward Facing Recommendations**

The project will hopefully have a legacy left in the inbound department. The instructions should be easy to pass down to new managers and PAs and I have also talked to the IT department to download whatever software that is necessary. In Amazon, processes are always changing which is why there are detailed instructions left to edit the code in what ever way needed as it was written to allow easy change without having too much coding experience.

The program is in no way perfect. However, it smoothens out the process and makes it much fairer. After seeing the process and learning how the inbound role staffing works, it was evident to me that this program was a much need addition. Due to the troubles, it has regarding FCLM website, it is limited at times. This is why SCC should be implemented into the entire inbound department soon (including stow). The dock and stow both lose time and money from SOS being done manually and unlike my program, the SCC has access to much more associate information as well as clock in times being updated faster than FCLM, allowing it to do what my program can do, but at a much more efficient rate.

**VII. Conclusion**

The program gives the inbound dock a much-needed roles rotation program which also speeds up the FastStart time, especially on heavy volume days. While it is far from perfect, the program accomplishes its goals and has the data to back it up. Getting the SCC integrated into the inbound department would be a great feature for the future, but for the time being my program does the job in saving time and the cost that comes with that time.

**VIII. Appendix**

Appendix A:

**A screenshot of a computer

Description automatically generated**

**A screenshot of a table

Description automatically generated**

Appendix B:

A screenshot of a computer

Description automatically generated

**A screenshot of a computer

Description automatically generated**

Appendix C:

A screenshot of a computer

Description automatically generated

Appendix D:

A table with numbers and a number of days

Description automatically generated