

Team name: KPMGComeGetUs

Name: Sng Hong Yao **Student ID:** 17205050

Name: Ankish Raj Prajapati **Student ID:** 17202456

Name: Ronan Mascarenhas **Student ID:** 17379773

Setting up Environment

1. Open a terminal and navigate to the file where run.sh is.
2. Execute `./run.sh` (Uses maven to build the JAR, then Java to run the JAR)

Meeting Demands

Storing a random candidate solution

- Done back in sprint 1, all students and projects are stored in Parser. A student object stores the projects being allocated to him/her, these include the project assigned, and the preference list. Nothing about the Student is stored in projects to simplify updating references (i.e. if we change something in Student, we need not update a Project).

Generating a random candidate solution (from data structures created in the previous sprint.)

- Again, done back in sprint 1, all students and projects are generated in Parser. Since Parser stores all projects (parsed from MiskatonicStaffMembers.xlsx), and it also stores all the names it can use (parsed from initials.txt), we can regenerate students and projects on demand whenever we call public methods in Parser (i.e. we generate it directly).