Project Allocation Portal

Albert Sunny, Sai Vamsi Alisetti, Subhash S

1 Introduction

Web Apps have been there since time being and with today's technology and opens source libraries there is a boost in the making of web apps. Our initial motivations of making this web portal was to reduce the drudgery people have to go through doing the project allocation manually in this modern era where everything is code driven. We have seen that most of the project allocations have been going on using Google Docs or Excel which is a very tiresome and time taking process. So we came up with an automated solution to this by making a web portal to collect everyones preferences and use an algorithm to allocate the projects. We will dive into the workings of the algorithm in the later sections. We made a web app as it is the most accessible thing these days.

2 Overview

We have four positions namely Students, Faculty, Admin, Super Admin. Every individual falls into any of these four positions. Each stream is called a program. As of now the programs are extended to pg programs as well.

Registraion & **Login**: Once they sign in with their institute email id we parse it and decide whether he/she is a student or faculty and redirect him/her accordingly.

Super Admin: Super Admin is given certain priviliges. She/he can view or delete all the projects, all the faculty, all the students, all the programs. Super Admin is the one who can add more programs by assigning it to a particular roll number map. Super Admin is the one who can set the status of admin to the faculty of the respective program.

Admin: Only faculty can be a admin to any program. Each program will have an admin. Admin is responsible for setting the deadlines of the allocation and sending mails to the respective members. Admin can also set certain other parameters like max number of projects per faculty, max number of student intake per faculty,total number of students per faculty, and admin can also view or delete students and faculty of the program.

Faculty: Faculty can choose to support more than one program by adding more programs in their profile. Each faculty has the eligibility to add, edit, delete projects and give a student preference order to each of their projects.

Students: Students page has a very minimalistic UI i.e it is very intuitive to use. All the students have to do is when the appropriate stage comes to pass they have to start choosing the projects they prefer.

Here we deal with taking preferences from two sides that is from the students (the projects preffered by them) and the faculty (the preference of students who applied to that particular project). Each project provided by the faculty have a student intake number i.e the number of students they wish to take for the project. Now we use the modified general version of Gale Shapley's Algorithm for stable matching of projects to students.

3 Algorithm Implementation and Development

Set S represents set of all students and set P represents set of all projects.

Let $S = \{S_1, S_2, ..., S_m\}$ where $m \in N$ and $S_i = (id_i, preference_i)$ is the i^{th} student and $preference_i = \{P_i, ..., P_j\}$ where $|preference_i| = |P|$ since the remaining projects which the i^{th} student didnt choose are sorted in the increasing order of number of students applied to the project, and have been appended to the i^{th} student preference.

Let $P = \{P_1, P_2, ..., P_n\}$ where $n \in N$ and $P_i = (id_i, preference_i, studentIntake_i)$ is the i^{th} project and $preference_i = \{S_i, ..., S_j\}$ where $|preference_i| = |S|$ since the remaining students who didnt apply to the i^{th} project are sorted in the decreasing order of CGPA and have been appended to the i^{th} project preference.

A necessary condition for this algorithm to work is $n \geq m$.

3.1 Pseudo Code

Algorithm 1: Project Allocation

```
Require: n ≥ m
  free ← S
  alloted ← [ ]
  allocationMap ← {}
  while |free| > 0 do
    S_i ← free[0]
    P_j ← first preference of S_i
    if ¬allocationMap.hasKey(P_j) then
      allocationMap[P_j] = [ ]
      allocationMap ← {P_j, S_i}
      alloted.push(S_i)
      free − {S_i}
    else
      if |allocationMap[P_j]| < P_j.studentIntake then
        allocationMap[P_j].push(S_i)
        Sort allocationMap[P_j] according to the index of S_i in the project preference
        alloted.push(S_i)
        free − {S_i}
      else
        S_k ← last element of the allocationMap[P_j]
        S_k,index ← findIndex of S_k in preferences of P_j
        S_i,index ← findIndex of S_i in preferences of P_j
        if S_k,index > S_i,index then
          allocationMap[P_j].push(S_i)
          Pop the first preference of S_k
          Sort allocationMap[P_j] according to the index of S_i in the project preference
          alloted − {S_k}
          free − {S_i}
          alloted.push(S_i)
          free.push(S_k)
        else
          Pop the first preference of S_i
        end if
      end if
    end if
  end while
```

4 Theoretical Background

Gale–Shapley algorithm provides a solution for finding stable relationships given a set of men and women. This algorithm can be used to solve a large number of problems, one such use case is matching interns to companies such that each company finds a suitable intern and each intern gets a desirable company.Suppose there are two couples (w1, m1) and (w2, m2). The relationship is unstable when w1 wants to be with m2 and vice versa. However, if w1 wants to be with m2 but m2 wants to be with w2 then it's sad for w1 but the relationship is stable. Please refer [2] for more insights on the algorithm.

5 Built With

MEAN STACK

MongoDB: MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schema. MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL).

Express JS: Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

Angular: Angular is a TypeScript-based open-source web application framework led by the Angular Team at Google and by a community of individuals and corporations.

Node JS: Node.js is an open-source, cross-platform, JavaScript runtime environment that executes JavaScript code outside of a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server- and client-side scripts

6 Summary and Conclusions

In this way we made the project allocation easier with the help of this algorithm and the web portal setup. To culminate this we have made many tests to check the efficiency of the algorithm and it works well with all the test cases.

References

[1] D.Gale, L.S.Shapley, *College Admissions and the Stability of Marriage*,
Available at `https://www.eecs.harvard.edu/cs286r/courses/fall09/papers/galeshapley.pdf`.

[2] *Applications of Gale-Shapley*,
Available at `https://medium.com/@UofCalifornia/how-a-matchmaking-algorithm-saved-lives-2a65ac448698`.