**Software Implementation**

We devided the complete project into smaller devisions to make things easier and to reduce the work load of which each member gets. We call this devision structure as '3D Structure'.

* Designing Team (Wireframes, UI)
* Developing Team (Flow Charts. Development)
* Documentation, Testing and QA Team (Reports, QA)
* Research and Development
* Development Tools -
* Github - We have used Github to use the coding collaborately to maintain time lines progress and to share the work progress of the useres.
* Google Drive - We used the Google Drive to share documents and Images (Reports and Charts).
* Trello - Trello is used to manage the project and assign the work for members
* Development IDE -
* NetBeans - We used the Oracle NetBeans IDE for development as we are using Java SE. NetBeans is a very user friendly IDE for developing Java.
* Requirement Analysis
* Work Breakdown Structure (WBS) - First of all we have had a discussion to identify the project domain to plan a project plan and the obstacles we may get.
* Gantt Chart - After the WBS, we made a gantt chart to make project development deadlines for development parts.
* UI, UX and Prototyping
* Wireframes - The Designing Team has made a project UI schematic as a basic UI development.
* High Fidelity - After disccussing and editing the wireframes for the requirement, the design team came up with a developed the wireframe to the actual design view.
* UI & UX Review - After coming up with a wireframe and High Fidelity designs we had make changes that we thought unneccesary. And also made changes for better UX.
* Architecture
* Flow Charts - After planning the project we have developed a Flow Chart to make the program work flow.
* Class Diagram - After making the Flow Chart, Then moved on to make the Class diagram which describe the structure of the system by showing its classes, their attributes, operations (or methods), and the relationships among objects.
* High Level Architecture - Divided into separate modules to work differently. Data Input, SA Solution Module, GA Solution Module and Results View Module. Each module work seperatelty to fullfill its intended work.
* UI Designing - For UI Designing we have used JAVA Swing and AWT libraries to make and develop the UI part of the project.
* Development
* File Handling - As an optional and additional feature, we have added a feature to load a data set (database) instead of hard coding it into the software.
* Develop Basic Mapping - An algorithm is used to map the projects to students to their most preferable selection.
* Implement Simulated Annealing (SA) - Simulated annealing is a probabilistic technique for finding a good (not necessarily perfect) solution to an optimization problem which is done by,
* Generate Random Solution
* Experiment with Temperature Schedules
* Define Energy Function
* Implement Genetic Algorithm (GA) - Genetic algorithm is a search heuristic that mimics the process of natural selection. This heuristic is routinely used to generate useful solutions to optimization and search problems which are,
* Generate Random Solutions
* Experiment with Population sizes, Mating & Culling Policies
* Define Fitness Function
* Define Invalid Mappings - Invalid mapping is when two students are assigned to a single project. This is resolved by Simmulated Annealing (SA) and Genetic Algorithm (GA).
* SA Solutions & Report Validity - When selected this option to solve the invalid mapping SA Solution algorithm runs 10 times to finds the best suitable solution. When this shows the solution we review and validates it as the best.
* GA Solution & Report Validity - When selected this option, this solves as same as the SA Solution but, runs n times to view the best suitable solution for us to validate as the best solution.
* Bug Resolving - After going through the test scenarios, we encounted some code errors and logic errors. We have resolved it by recodinng to go through again with test scenarios.
* Quality Assuarance - After developing the project to a nearly complete state, we have created and came up with Test scenarios to test the software with testing procedures. Which are Unit Testing, Component Testing, Integration Testing and User Testing.