



6CCS3PRJ Final Year Project Task Allocation System

First Progress Report

Author: Hussain Miah

Supervisor: Dr Tomasz Radzik

Student ID: 1114801

Date: 21st October 2014

Introduction

The idea behind the Task Allocation project is to have a database that models an organisation which holds data relating to employees and their skills and the projects that the company has considered to embark on. The database will be designed in a generic way allowing it to be used with multiple different organisation types where all the fields within the database may not be used by the particular organisation in question and they might also not be used during the matching algorithm.

After the user has imported their company information into the database they would then use the application to find an allocation for the employees that have been entered and the tasks that are available. The allocation that is presented to the user, will only be a suggestion and the manager will be able to save that information for later or do other processes within the work environment that need to happen.

This project has been split into two parts; database design and the algorithm that would be used to assign the employee to the task they are suitable for; for example an employee can only be assigned to a task when they have the right skills for that task and that they are available.

Database Design

The database will be designed for a generic company but when testing the application it will be assumed that, the company using the application will be a organisation within the technology sector. So when testing the application the database will contain technical terms such as Java, C# etc...

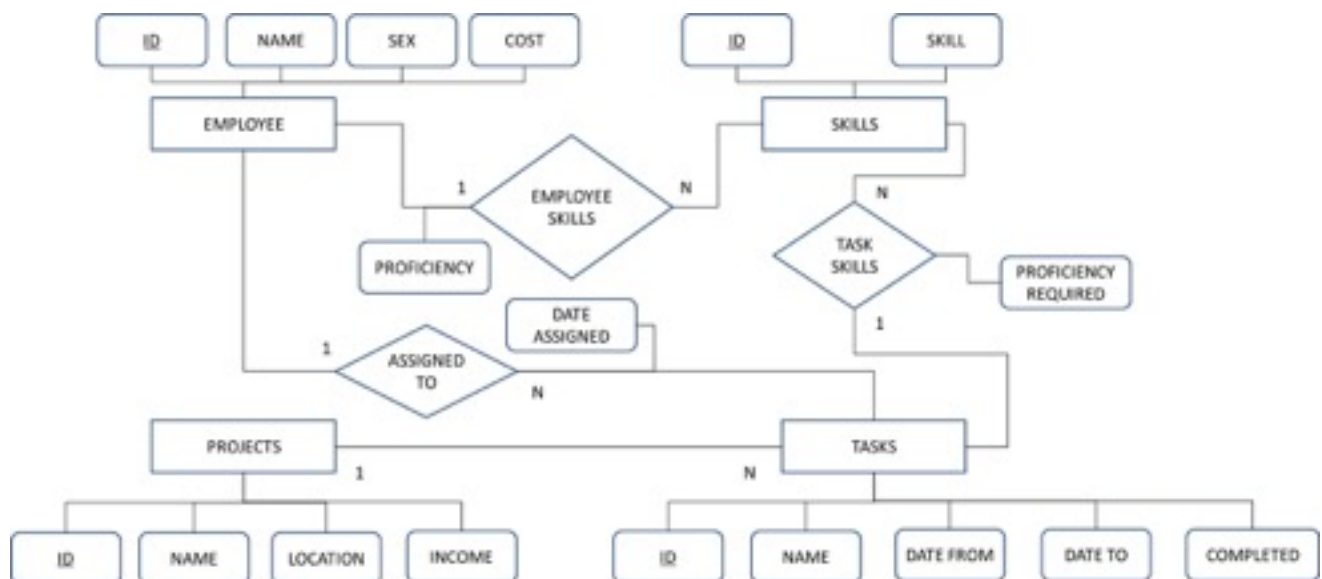


FIGURE 1: ENTITY RELATIONSHIP DIAGRAM OF PROPOSED DATABASE DESIGN

Figure 1 shows the entity relationship diagram for the proposed database design, there will be seven tables in total having the attributes defined in the diagram (underline denotes a primary key and a diamond represents a junction table; a table which links two other tables together in a one:many relationship).

The user will be able to import their data by using a CSV file and the application will read this file and insert the rows specified in the CSV into the database.

Allocation Design

When the user would like to see a possible allocation for the employees and tasks within the database I will first implement two ways of doing the allocation.

The first would be a generic greedy approach which will look at the list of all the employees and the tasks that need to be completed and the system will assign the first employee that meets the requirements for that particular task.

The second option is to create a bipartite graph of all the possible matchings depending on the skills that are required for the task and the availability of the employee, an algorithm will then be run over this graph to find the maximum allocation of the tasks. The way that I will be representing the graph within the code will be by using an adjacency list.

After the user has been presented with the allocation that is given to them, they will then have the option to assign these employees to the tasks that they have been matched up with and this information will be then reflected in the database.

Technical Specification

The application will be written in the Java programming language and I will be using the JDBC (Java Database Connectivity) API to connect with and query the database. The graphical user interface of the application will be written using Java Swing API. I will be using MySQL as my database architecture. There will be a script for creating the database, this script will also create the database user which will be used to query the database.

Functional Specification

The system will be able to provide the following features to the user.

- There will be a database schema that will be provided for the user to use with the application, which will hold all the information relating to the organisation. An example of this schema can be seen in Figure 1 and a more detailed explanation of the attributes of the tables can be found in the appendix.
- The user will be able to import data relating to their organisation into the database via a CSV file. As there are multiple tables that need to be populated at the start of the application, the user will need to use different CSV files, one for each table that is being populated.
- After the tables within the database has been populated with information relating to the employees, projects and tasks the user will then be able to start seeing suggested assignments of the employee to the task.
 - The user will be able to use a greedy approach to see a possible assignment of the tasks. This will assign the first employee that is available for the task.
 - The user will be able to see an assignment that maximises the profit of the project for the company where it will try and assign tasks to employees that not only have the right skill set for the task but also matching with employees that have a lower overhead maximising the profit that is achieved by the completing the project.
 - The application will be able to deal with employees that are already assigned to tasks previously, this is where the start and end date of the task will be used to determine if an employee will be available to take on the task.
- The user will be able to save the employee assignments that have been suggested by the application to a file, and they will also be able to open that file within the application to view the assignments.

- The user will be able to assign employees to the task that has been suggested by the application, they can either choose individual employees and the task that has been assigned to them or take the whole assignment set, this information will be stored within the database.

Appendix

Below are further details relating to the design of the database where the individual columns will be defined plus providing extra information where necessary.

EMPLOYEES

Attribute	Primary Key?	Foreign Key?	Comments
ID	YES		
NAME			This is will be split into the employees first and last name
SEX			
COST			The cost that is associated with this employee on a daily rate

SKILLS

Attribute	Primary Key?	Foreign Key?
ID	YES	
SKILL		

EMPLOYEE SKILLS

Attribute	Primary Key?	Foreign Key?	Comments
EMPLOYEE ID	YES	YES	This will act as a composite key
SKILL ID	YES	YES	This will act as a composite key
PROFICIENCY			There will be four levels: 1 - Beginner, 2 - Novice, 3 - Interimdate, 4 - Expert

TASKS

Attribute	Primary Key?	Foreign Key?	Comments
ID	YES		
NAME			
PROJECT ID		YES	
DATE FROM			This is the task start date
DATE TO			This is the task end date
COMPLETED			Whether the task has been completed or not

TASK SKILLS

Attribute	Primary Key?	Foreign Key?	Comments
TASK ID	YES	YES	This will act as a composite key
SKILL ID	YES	YES	This will act as a composite key
PROFICIENCY REQUIRED			There will be four levels: 1 - Beginner, 2 - Novice, 3 - Interimdiante, 4 - Expert

PROJECTS

Attribute	Primary Key?	Foreign Key?	Comments
ID	YES		
NAME			
LOCATION			
INCOME			This is how much the customer is willing to pay for the project

ASSIGNED TO

Attribute	Primary Key?	Foreign Key?	Comments
EMPLOYEE ID	YES	YES	This will act as a composite key
TASK ID	YES	YES	This will act as a composite key
DATE ASSIGNED			The date that the employee was assigned to the task

Gantt Chart

Below is the gantt chart to show the timeline of the project showing the major milestones within the project.

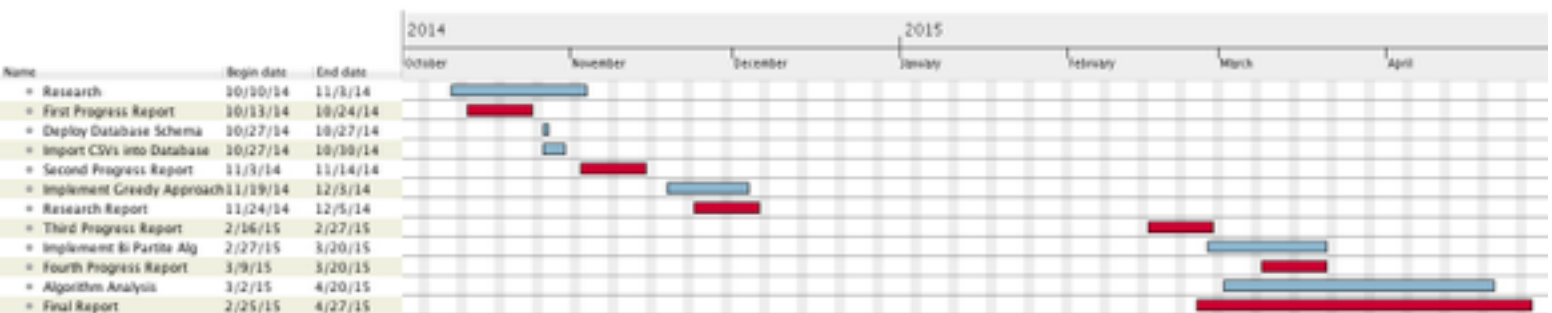


FIGURE 2: GANTT CHART FOR PROJECT.

Name	Begin date	End date	Duration
Research	10/10/14	11/3/14	17
First Progress Report	10/13/14	10/24/14	10
Deploy Database Schema	10/27/14	10/27/14	1
Import CSVs into Database	10/27/14	10/30/14	4
Second Progress Report	11/3/14	11/14/14	10
Implement Greedy Approach	11/19/14	12/3/14	11
Research Report	11/24/14	12/5/14	10
Third Progress Report	2/16/15	2/27/15	10
Implement Bi Partite Alg	2/27/15	3/20/15	16
Fourth Progress Report	3/9/15	3/20/15	10
Algorithm Analysis	3/2/15	4/20/15	36
Final Report	2/25/15	4/27/15	44