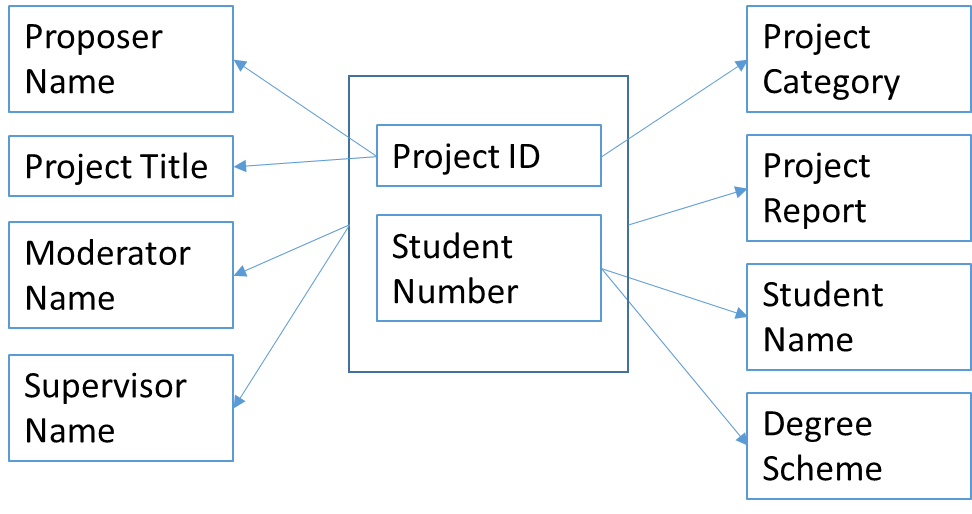
Database Systems Coursework 1

Q1.



Assumptions

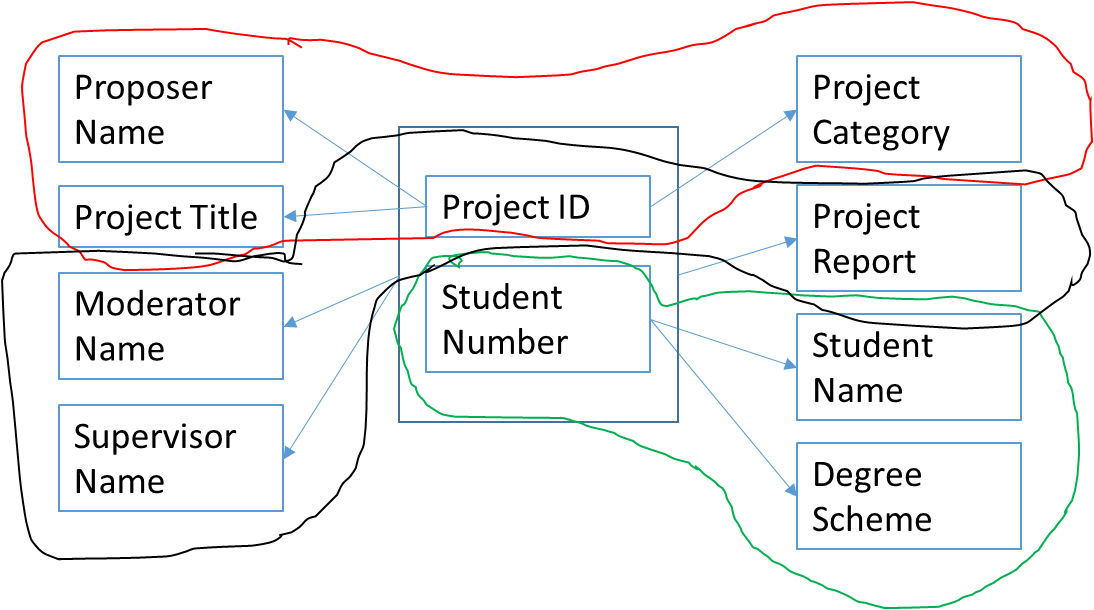
Staff names are unique

Only one student is assigned to each project

Multiple instances of the same project can be used by different students

Supervisors and moderators may be assigned to multiple projects but only one student

Q2.



Student

|  |  |  |
| --- | --- | --- |
| **Student Number** | **Student Name** | **Degree Scheme** |
| 0901202 | John Smith | BSc CS |
| 0934232 | Andy May | BSc CS |
| 0912436 | Alice Bell | BSc Cs |

Project

|  |  |  |  |
| --- | --- | --- | --- |
| **Project ID** | **Project Title** | **Proposer Name** | **Project Category** |
| 456 | Computer Game | John Smith | Student |
| 457 | Flight Simulator | Alan Turing | Staff |

SP

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Project ID** | **Student Number** | **Moderator Name** | **Supervisor Name** | **Project Report** |
| 456 | 0901202 | David Jones | Wendy Bell | To Be Submitted |
| 457 | 0934232 | Tom Lee | Alan Turing | Distinction |
| 457 | 0912436 | Mark Walker | Alan Turing | Merit |

Q3.

Spreadsheet designs are very demanding, both in terms of amending existing data and adding new data. If you were to add new data to this spreadsheet you would need to know the values for all ten columns which is very unlikely to be available all at the same time.

From the short extract of data we have available we can see that some students undertake the same project, if the project proposer were to change then this name would have to be changed in every place in the spreadsheet.

By contrast with the normalised design that I have proposed both of these burdens are somewhat alleviated. Splitting the data into three separate tables means that you need less data at any one time to completely fill a row of data.

In the case of the student table, we only need to know the student number, name and what degree scheme they are on, these three pieces of information in most cases will all be available at the same time and will most likely exist already somewhere else in the system, this way students can be added, amended and removed from the table without destroying associated data.

A similar case is true for the project table, projects can be added, amended or removed without having to worry about other data such as the student that will undertake them. This I believe also closer reflects the real life scenario where projects may be proposed from various sources at different times and added to a list that students can then choose from.

The project table also helps to cut out data replication, as we have assumed that multiple students have undertaken the same project, much of the data will be identical, and having the information for the projects stored separately means that only one entry is needed for the project regardless of how many students use it.

This then also solves the issue mentioned earlier of multiple amendments for a change of proposer name as the name only needs to be changed for a single entry in the project table.