ASSIGNMENT 1 – REPORT ELEMENTS

1. **Group number and Group Members**
   12 – Stefan, Baderin, Olaoluwa, Alex

2. **Introduction –** (what the report is about, who the design is for etc)

   It's about the Academic research administration system at the North London University. Basically the administration system is not clear. It is rather messed up and needs the academic research process to be put into a database to make the system clearer. The design is for the university's School of Computing department who wish to set up a database to keep track of the academic research process. It also for different people who will be involved with the database in some way such as the university staff, students and people from outside the university such as guest lectures.

   Assumptions made

   The case study is a brief description of the scenario and database requirements and may contain ambiguities which an analyst would normally clear up by referring to an end user for clarification. However, your group does not have access to any end users, so you need to make assumptions instead. These assumptions should be things that will affect the design of the database. For example, the scenario contains the statement '**the university would like to monitor attendance at each of the student lectures'.** This could be interpreted in many ways, so you might want to clarify this, eg:

   - "We assume that monitoring attendance means counting the total number of students who attend and recording this number in the database." (In which case, this might be shown in your design as an additional attribute in a 'lecture' entity)

   Or

   - "We assume that the monitoring attendance requirement means that the ID of every student who attends the lecture must be recorded". (In which case, your design might contain an additional 'StudentAttendance' entity)

   Note that you should be aiming to provide a finished design that is as realistic as possible, so providing assumptions that are aimed at simplifying the design is unlikely to produce a good mark.

3. **Stakeholders**

   Identify and briefly describe 4 stakeholders - ie people/roles who have an interest in the database – ie will be interacting with the database in some way. eg students

   Researchers – they study the papers or journals written by academics…

Academics – they write the academic papers. They will provide information about the academic papers written by them which will be put into the database.

Students -

Dean of the School of Computing

4. **User Requirements (3 per group member)**

Functional requirements are what the stakeholders will be expecting from the database – ie specific reports or queries etc. You should express them in the form: ***"The system must be able to ……"*** and try to link them to a specific stakeholder, eg:

Student:
R1: The system must be able to display a timetable of forthcoming lectures, showing the date/time, location and topic of each lecture.
R2: The system must be able to …. etc.
.. and so on for your other stakeholders. If you cannot think of three decent requirements for a single stakeholder, then you can suggest a couple for other stakeholders.

Students:

R1: The system should be able to show details of the forthcoming lectures. This must show when and what time the lectures will be starting and what the research topic and or research project is going to be, for every lecture. It should show who the researcher is, that is going to be delivering the lecture, the course and or the module. This should be like a timetable.

This information could be put into a table called Guest lecture or Research lecture. It may have columns labelled Lecture/Lecture description, Course name or Module name, Guest lecturer, Researcher assigned, Guest researcher, Research interest/topic/Research project, Lecture Date and Location.

R2: The system should be able to show, for each course and or module, the students who are enrolled on the course with their ID no., the date of the lecture and whether they attended or not which could be shown as a Boolean data type – Yes/No or Y/N.

This information could be put into a table called Student attendance or Guest lecture attendance. It may have columns labelled Lecture date, Location, Guest lecturer, Researcher Assigned or Guest researcher, Lecture description, Course name or Module name, Research topic/interest/Research project, Student name, Student ID number and Attended (Yes or No)

…The system should be able to show which students attended the lectures and information from other tables…this is how the attendance of the lectures will be monitored…

Dean of School of Computing:
R1: The system must be able to keep track of the articles that have been published. This includes who has published the articles, when and where they were published and which project they are related to etc. It must show the names of the academics or authors who have published the articles, names of the related projects and the location they were published in.

R2: Other information it must show is where the papers were written in a date format. The system must have all this information listed in a table with columns each labelled the name of the publisher, *the location published, the date published and the related project. The dean of school requires the system to have all this information stored in the database. He/she is going to be getting this information from the system.*

Academics:

R1: Information provided by the academics about the papers they have written will be fed into the system. The system must be able to store all the information such as who completed the articles, which project the article is about, when and where it was completed is going to be provided by them.

Researcher:

R1: *The system should be able to keep a record of what each of the researcher's topic or research project will be, when and where they will deliver their lecture and which course and or module the lecture is for. All this should be put into a database table so that it can be put into a report for the dean of the school.*

5. **Entity Relationship Diagram**

   Here, paste a copy of you ERD. The ERD should show the entities, attributes, primary keys, foreign keys, and relationships. You can draw your diagram using the Microsoft Office Visio drawing tool (or you can find other (better) freebie tools on the internet and use those if you wish). Please produce your diagram using the Crow's Foot notation, **NOT** the Chan notation.

6. **Entity Specification Forms**

   You should include a form for each entity shown in your Entity Relationship diagram. These forms make up your data dictionary – ie they are the prime source of all knowledge to do with your database and should describe all the attributes without any ambiguity. There should be enough detail and clarity to allow another group to take your forms and implement your design without asking for any clarification.

   Please see the example of the entity spec form for further fuidance.

GENERAL NOTES:

The 'product backlog' is a live document that shows who is doing what. It is a way of structuring the work that needs doing to complete the assignment, and is revised on a weekly basis. The idea is that you break the assignment town into tasks, allocate tasks to individuals, and reviewed progress on a weekly basis. The example provided can be tweaked, but the main elements – who is doing what and whether they have done them must be shown. Note that subsequent tasks can be tweaked depending on what have been done – eg your group may decide they want to speed up tasks, add further tasks etc.

You can include other tasks that are indirectly related to the assignment – eg doing research on an alternative package for drawing the ER diagram, or having time to research some aspect that needs further research, or even for contributing to a social activity (… as a team building exercise!).

The latest version of the 'product backlog' spreadsheet must be uploaded to Blackboard on a weekly basis.

Note: the scenario can be broken down into a number of areas of functionality:

- The placement admin system
- The guest lecture system
- Keeping track of external speakers
- Etc

Initially, you might find it helpful to allocate a particular area to a specific team member to analyse – then gather all the bits together for a review by all team members.