# Module CO550WBL Web Applications

## Assignment CW1A – Team Project Design

Student names:

* Natalia Fernández Pintos (Student ID: 2191814)
* Wafa Zaidan (Student ID: 22000104)

1. The project idea

This project consists of a web application for the teachers and students of the Digital and Technology Solutions degree. The application provides a user interface in which the users can visualise a calendar that displays a series of events related to the modules of the degree. Users need to create an account and log in to gain access to the calendar, where it is possible to visualise the existing events or create new ones. Furthermore, the application also features the possibility for users to easily edit and delete events that they have created. There is also a profile page for each user, in which they can access and edit their personal details, as well as view a list of the events they have added to the calendar, filtered by month and year. This list provides a simplified view of the main details for the events, as well as an option to visualise all the information for a single event and update any details.

1. Background

The idea for this project originates from the need of a unified location in which to easily display information about the timetables and deadlines for the Digital and Technology Solutions degree, as well as details for the attendance to the onsite lectures. Currently, the information about the timetables and deadlines for the modules of the degree is available in a module guide file, and attendance is communicated via different messaging channels (social media or email). The disadvantages of the current methods arise from the difficulty of updating the timetables document, as it is recorded in a static file and would require the teacher to create and upload a new one each time there is the requirement to do any changes to it, and additionally, it would need every student to download a new copy of the document. Furthermore, keeping record of each time a student or teacher is unable to attend a lesson can also be quite inconvenient.

With the web application proposed in this project, the aim is to solve these problems by developing a data-driven solution in which teachers and students can keep all this information in a single place. In the first place, the web application would enable its users to keep track of all the information related to the different modules of the degree in a simple way. Secondly, any changes to the existing schedule would be effortless and instantaneous, improving the current need to work with static files. Lastly, the application would enhance the communication between teachers and students, providing the option for users to notify others of any changes in the schedule.

1. Using web applications

The reasons why a web application is suitable for this project are many. In the first place, creating a web application to give access to the schedule information provides availability of this data in a single place with only a couple of clicks, and across a variety of devices. It also has the added benefit of allowing the users to interact with the information in the website in an easy way, which is something that is expected in websites nowadays, as explained by Offutt (2002) we now refer to the visitors of a website as *users*, implying interaction. Moreover, a web application is extensible, and therefore allows the solution to grow and meet new requirements that can arise in the future. Additionally, using a web application that is data-driven has the advantage that we only need to install a web browser to access all the information and services that it provides (Richardson and Ruby, 2007, p. 2), therefore, removing the necessity for each person to install their own calendar applications, add and maintain updated the data across different users manually.

1. Data management:

As the primary objective of our web application is to facilitate timetable management between students and teachers at Bucks university, we recognize that we will be dealing with large amounts of data that need to be well managed and stored, easily accessible and highly maintainable.

For that reason, we made sure to follow a process of data modelling where we identify and determine real-life information and put them coherently into a well-formed data structure. This process is essential to determine and analyze the kind of information we need in our application which means we can get a clearer understanding of the exact data that we need early-on (Domanski and Irvine, 2000, p. 28).

The first step in our data modelling process was to build an Entity relationship Diagram which is an abstract representation of the data that will be used in our program. The ERD format has three main elements; entities, relationships and attributes (Liu, 2011, p. 276).

In our program, we defined two main entities which are the `User` and the `Event` as the application mainly revolves around users creating events using our calendar and to share and notify other users with key events.

Then we identified the attributes that each entity will have along with their data structure. For example, for the User we included an Id and a role Id of type integer, and a username, password and email of type string. Whereas for the Event we identified an event Id, user Id, category Id of type integer, start-date and time and end-date and time of type datetime, a title, description of type string and finally a notify field of type Boolean.

Then we moved on to define the type of relationship between them and the multiplicities of each. For example, a user can create, update or delete zero to many numbers of events and an event can be managed by only one user.

After creating the ERD, we moved on to building our relational-database schema where we specify the description of the structure and behavior of our database. (Balsters, 2000, p. 1). This is an important step where we visualized the entities that will exist in the database in the form of tables with their specific fields and their relationship with each other, whether they have a One-To-One, One-To-Many or Many-To-Many relationship (Gupta, 2007, p.15).

Building upon the ERD we had, we identified four tables which include users, events, categories and roles and depicted the relationship that they will have for each other. We also listed the fields that each table will have and marked those that will constitute the primary or the foreign key.

This process of thinking about how to manage the data that will be used in our application has helped us understand what data we need to build it, and how it will be structured and stored. It has also prompted us to think about the main features that we will provide in our application and what sort of data we need for each feature.

Finally, we strongly believe that this process has provided us with solid foundation to build upon in the next stage where we start coding to build our calendar application.

1. References

Offutt, J. (2002) ‘Quality attributes of Web software applications’, *IEEE Software*, 19(2), pp. 25-32. doi: 10.1109/52.991329.

Richardson, L. and Ruby, S. (2007) *RESTful Web Services.* Sebastopol: O’Reilly.

Domanski, P. and Irvine, P. (2000) ‘A Practical Guide to Relational Database Design’. Herefordshire: Diaxon ltd, pp. 28

# Liu, H. (2011) ‘Oracle Database Performance and Scalability: A Quantitative Approach’. Wiley, pp.276

Balsters, H., Brock, B., and Conrad, S. (2000) ‘Database Schema Evolution and Meta-Modeling’**.** Dagstuhl Castle: Springer, pp.1

Gupta, K. (2007) ‘Taxonomy of Database Management System’. New Delhi: Firewall Media, pp.15