

# Assignment 2

## Management System

**ZEIT2105 System analysis and design – S2, 2020**

Never Stand Still

School of Engineering and IT

### Due Dates

Notification of groups and roles:  
25<sup>th</sup> September, 5:00pm, 2020

Deliverable submission deadlines:

- Final submission due Friday 4<sup>th</sup> November, 2020.
- Presentation
- All presentations must be complete by 23<sup>th</sup> of October

### Weight

This assignment contributes 30% to your overall mark for the course.

### Assignment Objectives

This assignment provides you with the opportunity to demonstrate your system and database design skills, as well as your skills implementing a SQLite database.

### Groups

This is a group assignment. Groups should be 2-5 members. You should aim for a mixture of skills in your group as you will need members to specialise in design and implementation. You must email me ([t.lynar@adfa.edu.au](mailto:t.lynar@adfa.edu.au)) nominating:

- the members of your group
- the roles they will play

You must use git to show your teamwork. Store all the artefacts that your team produces in git. How you use git is up to you but you must provide me with a copy of your repo when you submit the assignment.

### Your Task in Brief

You are working of a new cyber education start-up. They specialise in micro credentials. The curriculum is accompanied by a system of 'badges' that is given out when students complete certain groups of activities. Teachers need to keep track of which students complete which activities so that they can receive their badges in a timely manner.

Currently, the start-up does this using a (paper) notebook in which it records attendance each week and jots down which activities completed. Each week we have to update the achievement record from our jottings in the attendance record book.

In this project you will design and implement an attendance record system that helps teachers manage student attendance and achievement records by storing details in a SQLite database.

### Submission Requirements

You must submit a softcopy of your code (Task 3) by email and hardcopy of your design (Tasks 1&2). You may submit your design as early as you like (week 12 recommended) up until the final due date of 4<sup>th</sup> November.

### Detailed Statement of Requirements

The micro credential award scheme is fairly complex. There are loads of things people can do. Students attend weekly meetings, weekend activities, and also do work in their own time if they want to. There are Achievement Badges that let students focus on a special interest or skill and badges that focus on the core curriculum. There is also a leader board that shows those students with the highest number of credentials.

The badging system has three levels gold, diamond, and super platinum

Each badge has 14 Tests students must complete all of Tests 1-7 and any 3 of the remaining Tests (8-14). Each Test 1-14 is broken into topics. Each topic has between 1 and 3 parts that must be completed for the topic to be considered complete. We only need to record the date that the entire topic has been completed, but sometimes we do parts of topics and we thus need to record completion of a part as well (although not necessarily the date). In some tests students do not need to complete all the topics to complete a test. Rather students or teaching staff may choose 2-3 topics of their own preference out of the test.

### Key Functionality and Scope

There are a number of things that we do manually each week that we'd like to automate using an app:

#### *Marking attendance:*

Each week during trimesters 1, 2, and 3, students attend an hour and a half meeting. Sometimes students are sick or otherwise unable to attend. We need to keep

a record of which students attend which weeks. We'd like to be able to mark attendance using an app that stores weekly attendance data in an SQLite database. Ideally, in future it would be great if there is a single database.

### Recording tests

At most weekly meetings, teaching staff run a program that satisfies the requirements for parts of test topics, whole topics or several topics comprising a test. Some weeks we choose not to do related work. Sometimes we hold weekend intensives and do parts, topics or tests at the intensive. We need to be able to record what we did and when.

## Your Task in Detail

Your assignment comprises four sub-tasks for requirements gathering, design, implementation and administration:

### 1. Requirements List:

#### Task 1 Deliverable

Submit a list of requirements. Prioritise the requirements to indicate, which will be covered in your system design and implementation and which will not.

### 2. Design:

- Design a database that can store at least information about
  - students
  - completed parts, topics and tests.
- Design an app with the following minimum functionality:
  - Add a student to the database, including *at least* their first name, last name, DOB and which group they attend.
  - Record attendance for a weekly meeting or weekend intensives.
  - Record parts, topics or tests completed.
  - Search by name for a student and display their personal details and achievements
- Other functionality you may like to consider:
  - Search for students by other criteria eg: age
  - Notify teaching staff when a student has completed an entire badge (7 compulsory tests + 3 optional tests)
  - Notify or permit teaching staff to search for parts, topics or tests that have not been widely completed (and thus should be the focus of an upcoming meeting program)
  - Edit details
  - Enter a planned test and the name of the person who will run it.
  - A schedule

- Any other requirements you can establish

#### Task 2 Deliverables:

- A **use case diagram** showing the functionality proposed for the new system.
- A **relational database schema** showing the proposed structure of the database
- An **activity diagram** showing the app behaviour for the key use cases.
- A **class diagram** showing the app structure. You need only diagram your classes, not system generated classes.
- A **windows navigation diagram** showing how all the screens used by the system are related and how the user moves from one to another.

### 3. Implementation:

- Implement an app that communicates with a local SQLite database to achieve the following minimum functionality:
  - Add a student to the database, including at least their first name, last name, DOB, which colour six they are in.
  - Record attendance
  - Record parts, topics or tests completed at a weekly meeting or weekend workshop.
  - Search by name and display their personal details and achievements

#### Task 3 Deliverables:

- Your App Project code by Friday 4<sup>th</sup> November, 2014.
- A presentation and demonstration of your app functionality (time by appointment).

### 4. Administration:

#### Task 4 Deliverables (Individual Submission):

- Each group member must submit a signed statement of how they would like me to divide the 100 marks for the project between group participants. Eg: if you believe that every group member contributed equally, then advise me to divide marks equally. If you believe that one or more group members did not contribute equally to the project, you may advise me appropriately.
- Note that contributions may be made in different manners, but still weighted equally (eg: one person may contribute to design, while another contributes to implementation).