Programming Project

Alex B

StudySupport

Analysis

# A1: Problem Definition

## Description of client

My client is a head of year 12 for St. Wilfrid’s RC College which provides education for students from 11-19 years old. My client has 2 classes and is head of her department. She will need to access the data of the students in her classes and the data of students in the entire year so she is able to monitor their progress. Primary Stakeholders of such a project would include members of staff such as teachers and students with parental access if necessary, teachers would have access to students’ results, deadlines and the upload section but would also have administrative abilities to view all students’ scores in their class, additionally being able to give feedback to students. Secondary stakeholders would be the government and local community who may wish to implement the system in other schools with its adaptability.

## Problem description

The problem is that my client is using a paper based feedback system, which is hard to monitor and track the academic results of individual students. Another issue is that students are given very generic feedback because of this paper system. My client also uses a spreadsheet based tracker which is difficult to interpret and setup and provide feedback to students with. My client would like an application that is easy to use for new members of staff and have multiple access levels for the head of department, teachers and students. Not only this but my client has requested that multiple users can use this system at once, without overwriting each other and the different levels of access with have different permissions based upon their access level.

## Current methods used by Stakeholder/End user and areas of improvement

Current method of use:

My client currently uses a paper and spreadsheet based tracking system. This system has no student involvement and feedback is often given on paper sheets which can be lost very easily. The tracker system is not very user friendly and is difficult to use for new teachers. The tracker is colour coded with red for below target and no other colours. This is very basic and I would like to expand the use of colours with orange for on target, and green for above target, which would give a more detailed overview of how students are progressing.

Area of improvement:

An area that I am looking to improve for my client is the user-friendliness aspect of the tracker and the access to the tracker for students. Being a desktop application every student will be able to access it via a school computer. The tracker will also use multiple levels of access for both students and teachers providing feedback and a deadlines section for students. I intend to link together all data about assessments for students and teachers so that it is more convenient than having to access multiple programs or sheets of paper, in which files could be misplaced or paper lost.

# A2: How is this problem amenable to computational methods?

Thinking abstractly and Visualisation

When thinking abstractly I would have to analyse what data is necessary for the problem to be solved and what data does not have to be included so that the problem is clearer to understand. The students name, candidate number, assessment scores, year group, class and subjects would be necessary pieces of data for the tracker to operate. Unnecessary pieces of data would be items such as date of birth, timetable and pictures of students. This data would not be necessary for the tracker to work and would overly complicate the tracker. Graphs can be used to analyse the mean average of the results in the class and identify patters in the data.

Thinking ahead

Thinking ahead involves planning inputs and outputs; this is useful as I can ask what answers I need before I start. Data inputs such as the student’s assessments scores, name and class that they are in so that they can be fed into the database. This will be required for the solution to work as these inputs are necessary to track and identify students.

Thinking procedurally and Decomposition

The problem of an assessment tracker can be decomposed into numerous more manageable sub problems, these include: Deadlines, upload area, previous assignments and feedback from assignments. There are already solutions for some of the sub problems such as an upload system that can be incorporated from a library into my final code. Additionally, a calendar system has already been created, which can be adapted into a deadlines system for students.

Thinking Logically

The problem would have repetition in the completing the loop aspect of the system. The school will set approximately 5 assessments per term thus the system will be repeated 5 times. Firstly, the student will be set the task by the teacher and add it to their deadlines, once completed, the student will upload the task. The teachers will then mark the assessment, input the scores into the database and give feedback to the students. The cycle will repeat for every assessment. The system will also branch as given opportunities based on the scoring system of the assessments. Depending on the students score, the program will either mark the student as: Below target grade, on track, and above target grade.

Think Concurrently

An example of thinking concurrently would be that different users will be able to access the database at the same time; however, the database will not allow for concurrent user access to avoid data redundancy. This would take effect when multiple teachers are accessing and updating records for the same student. Furthermore, the upload section of the application will allow for multi-user access to uploading files without files being overwritten. This will be incorporated into the student upload section, to which teachers can access the student’s assessments and will be notified when assessments are uploaded.

# A3: Research

## What I intend to research

I intend to research similar applications that have been developed based upon assessment trackers. From this I will be able to take inspiration on what features work and what features are unnecessary for my application.

Exact details of how the current system paper/ computerised works

Id strengths and weaknesses