

Criterion A

Client and Problem

The client is Ms Liew Kwai Yoke, also known as Ms Mcloughlin. She is an IGCSE and IB Mathematics Teacher and the head of the Peer Tutoring Club of the school. Her current roles as head of the club are managing the various Peer Tutoring sessions and club, recruiting new Peer Tutors and selecting Peer Tutor leaders.

According to Ms Mcloughlin, the system she has for the club's organization is inefficient and wastes a lot of time (Mcloughlin 2022). This is because it's a simple spreadsheet system, in which she has to manually enter all the data from the surveys tutors fill out at the start of the semester. The system is manually operated, resulting in continuous self-organisation of the system for the client.

Name	Grade	Days	Subjects	Tutee		
1 AGARWAL Savit	12	M, W, Th	HL Comp Sci, HL P, HL AA, SL Econ			
2 CHEN Wu Chi	12	Th	SL Chem			
3 CHOI Chungwoo	12	T	IB Eng	CHOI Seungwoo		
4 CHOI Seungwoo	12	W	SL/HL? Chem			
5 CHOI Seungwoo	12	M	P	CHOI Chungwoo		
6 DAYAL Aryan	12	Th	HL Chem			
7 FANG Ruibing	12	Th	SL Chem			

Figure 1: Current System for Peer Tutoring Organization

Furthermore, when pairing a tutee with a tutor, she has to look for a tutor in this spreadsheet based on their grade, day of availability, and subject tutored manually, which is a highly inefficient process (Mcloughlin 2022). Usually, there may be multiple tutors that fit the criteria of the tutee. In that case, there is no method to distinguish between the tutors; she just has to choose one of them. This can be a problem as she doesn't know who to choose, so there is randomness in selecting a tutor.

Therefore, the current "system", which doesn't exist, leads to an inefficient, laborious and random process of organizing Peer tutoring.

Word Count: 241

Proposed product

As a leader of the Peer Tutoring club, it took me aback that Ms Mcloughlin was spending so much time organising something that should have been relatively simple. This fact justifies a solution to make the system more efficient.

My solution is to produce a Menu-driven system that will allow Ms Mcloughlin to efficiently manage the different Peer Tutoring sessions, the tutors and tutees, and the subjects tutored/studied in the club. The system will rank tutors after Ms Mcloughlin enters the details of both tutees and tutors based on the available sessions, the subject grade, and the level (HL or SL), making the system automated for the client.

This ranking will be based on the feedback ratings and past reviews of the tutor from tutees. Based on this information, the system will rank the Peer Tutors available, allowing Ms Mcloughlin to make an educated decision about which tutor to select to tutor the tutee. Furthermore, the system will also automatically store and manage the changing nature of tutors and tutees using serialization and deserialization of objects, simplifying the client's work.

Therefore, the system is much definitely better as it provides a method for the client to distinguish between tutors, allowing her to easily pair tutees with tutors based on their needs and experience in a tutor. It also provides much more efficient access and storage of data.

Word Count: 229

The system will be programmed on IntelliJ IDEA 2022.2.2 (Ultimate Edition) on the operating system macOS 10.14.6 with Java version 14.0.1.

Specific performance success criteria

1. The client must be able to enter, edit, and delete the tutors' and tutees' names, grade levels, subjects taught/wanting to learn and session availability.
2. The system should automatically save the data entered by the client to prevent the data not getting deleted every time the system is shut down.
3. The system should be able to rank the tutors based on the feedback received from the tutees and allow the client to influence the ranking.
4. The system should prevent the input of data that fails the validation checks, is entered in an incorrect format, or is entered incorrectly, and handle these errors effectively.
5. The client should be able to obtain the information of any tutor and tutee by searching the system using their name.
6. The client should not have to instantiate the ID of any person in the system.
7. The system Java Application must be less than 500MB in total.