Sprint 2

Planning

      Second sprint's plan was to create a database using SQL to hold the data of the students and the date and location where they were detected not wearing a mask. Additionally, a new feature was to be added which is the ability to book an antigen test from within the app by filling a form for that. Finally, it was planned to create a prototype for our application to showcase its features.

Sprint Backlog

ID	User Story	Tasks	Owner	Status	Estimated effort%	Day 1	Day 12	Day 13	Day 14	Day 15
					SPRINT [2] BACKLOG					
1	As a security, I need the system to detect any student who is not wearing a mask	Design an ER Diagram and Relational Model to be used for creating the database.	O.Abdelbaset	Completed		0	5	5	8	2
	As an AUC member, I want to send a petition if my face was detected mistakenly.				40					
	As an administrator, I need to verify validate the petition of the petitioner	Create the database system to keep track of vioation records and petitions as well as access requests.	S.Sakran	Completed		0	5	5	7	3
2	As a tester, I want to test the system and make sure it is working properly	Database should be tested with any random or fake data.	O.Moawad	Completed	20	0	5	5	4	6
3	As a student, I do NOT need my face to be detected while eating/drinking or changin a mask	Design and refine an OOD that shows the sequencial process and any threats to the efficiency of the system.	K.Aboukoura	Completed	20	0	5	5	5	5
4	As an AUC member, I want to take antigen tests whenever I have symptoms	Develop an antigen test module to allow requests and regulate the process.	O.Fayed	Completed	20	0	5	5	10	0
					100	0	25	25	34	16

Status Review

      A database was developed using SQL for the students name and ID, date and location of detected students to save their details to apply fines. A c++ code was developed for the form that will replace the daily selfcheck with an antigen test booking and we would develop a database also for the time slots in future sprints..

Retrospective

What went well: An Entity Relation Diagram and Relational Model were designed to create a database system using MySQL. The antigen test booking feature was implemented as well. Each entry in the database represents an incident of detecting a member in the community not abiding by Covid-19 regulations indicating the name, ID, date, and location. AUC members who were found to be violating the regulations can apply for petitions to reconsider their case, and an administrator is to validate their case as either 'A' (referring to accepted) or 'D' (referring to declined). Also the self-check process has been developed to make our lives easier by automatically granting access to campus if the measured temperature meets the acceptable range as an indicator. While applying for antigen test booking, the student indicates whether he was in contact with a positive case or not, and if yes, they should report the name and ID of that case if they were AUCian.

What could be improved: Finish and deliver the final prototype for our system by implementing the systems we developed in C++ code. Add more database systems to cover more data such as the bookings of the antigen tests. We could have been more effective in choosing a platform and start to learn it to help us deliver the prototype on time. Also we could have searched for actual hardware devices to implement our systems but unfortunately our systems need advanced hardware devices that are not commonly available for students.

What should be done in the next sprint: developing a prototype for the application using any sophisticated platform by using all the work done in all sprints including diagrams, C++ codes and database into delivering a final prototype concluding our systems in the project. So our team during the next sprint will search for a platform that can help us implement the prototype and to be a user-friendly platform to allow us to learn it easily.

Entity Relation Diagram
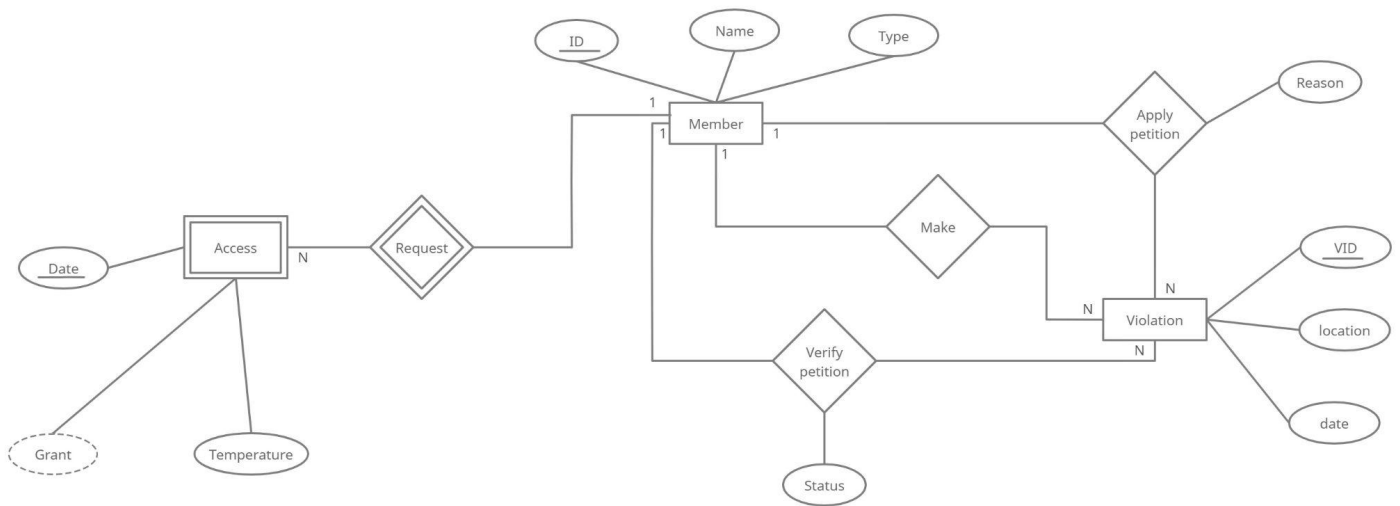
Relational Model

```
Member(ID-PK, Name, Type)
Violation(VID-PK, location, date, ID_Make-FK)
Petition_Violation((ID_Petition-FK, VID-FK)-PK, Reason)
Verify_Petition((ID_Verify-FK, VID-FK)-PK, Status)
AccessRequest(ID-PK, Date-PK, Temperature, Grant_Access)
```

MySQL code for creating the database

https://github.com/Abdelbaset65/CSCE3701_Project

https://github.com/Abdelbaset65/CSCE3701_Project/blob/main/Database%20Code.sql

Demos

https://drive.google.com/drive/folders/1n-NIqQYN1ZBJU47vD46d9ATlu0ewG8su?usp=sharing