

Student Portal

Local schools often struggle to provide students with easy access to their academic information and co-curricular activities. In many cases, students lack a centralized platform where they can view essential academic information and transcripts of their participation in co-curricular activities. This absence of digital tools hinders students' ability to stay informed about their progress and involvement in school activities.

Additionally, students may find it challenging to track their academic achievements and co-curricular involvement, which are crucial for personal development and future opportunities. Without easy access to this information, students may miss important deadlines, fail to recognize their accomplishments, and lack the motivation to engage in school activities.

Moreover, students in local government schools often have limited exposure to technology and digital learning tools. This not only hampers their learning experience but also reduces their digital literacy, which is crucial for success in today's world.

Project Introduction

The Student Portal project aims to address the challenges faced by the majority of local schools by providing a comprehensive digital platform that enhances communication, access to resources, and overall educational engagement. Our project aligns with the goals of promoting inclusive and [equitable quality education \(SDG 4\)](#) and fostering lifelong learning opportunities for all.

Ultimately, the Student Portal is designed to support the digital transformation of local schools, ensuring that students receive a high-quality education that prepares them for future success.

In this project, we will only target the 2 essential functionalities:

1. **Academic Page**, where students can view the informations of their classes
2. **Co-curriculum Page**, where students can view their participated co-curricular activities, generate transcript and track their [co-curriculum marks \(PAJSK\)](#).

What You Will Need To Do

We separated our project functionalities into basic features and extra features. The basic features are the main basic functionalities that our project must have in order to make it workable, the extra features on the other hand will be the additional functionalities which improve the project overall as a whole, which includes scalability and accessibility.

Note that GUI (Graphical User Interface) is considered an extra feature, you may decide to use the CLI (Command-Line Interface) as the interface for the end users to interact with the software.

Basic Features (8 Marks)

User Account [1 mark]

Each user will have their own user account. Therefore, you will need to create a User class which contains the following required fields:

- Student Email
- Matric Number
- Password
- Academic Subjects
- Co-curricular Clubs

Data Storage [1 mark]

Throughout the project, we expect various data generated like user details, user's current state and many more. So, we will need to store all related data in data storage so that the data remains saved even after the program is terminated. For basic features, you may save each user's data in CSV, TXT or BAT format.

Note that using external data storage or databases (*refer to Extra Features*) are counted as extra features. However, you are still required to have **AT LEAST ONE** file I/O application in order to get this mark.

Login / Registration Page [1 mark]

Since our project is user-based, we will need a sign-up page for the users to register themselves and a login page for users to log into their accounts from any location. To ease yourself in logic management, the user will log in only by using a valid email, instead of the option to log in with a username.

Here, I have created 2 mock user accounts for you to test your program out, you can create more accounts if you want but this 2 accounts have to be in your project for this section's presentation.

UserData.txt
s100201@student.fop s100201 pw-Stud#1 1103, 1119, 1225, 1249, 1449, 3472, 4531, 4541, 4551, 6531 B07, P82, S01 s100202@student.fop S100202 pw-Stud#2 1103, 1119, 1223, 1249, 1449, 1511, 3754, 3766, 3756, 3767 B01, P27, S15

The list above is in the format of StudentEmail, MatricNumber, Password, AcademicSubjectsList, CocurricularClubsList, separated by a "New Line" or `\n`. Copy and paste the list above in **UserData.txt** file, meaning that the content of that file should look **EXACTLY** as the content shown above.

Academic Page [1 mark]

As described in Project Introduction, this page shows the academic subjects that is enrolled by the student. To get things started, take a look at the list of academic subjects provided by the majority secondary schools in Malaysia:

AcademicSubjects.txt

```
1103,Malay Language
1119,English Language
1223,Islamic Education
1225,Moral Education
1249,History
1449,Mathematics
1511,Science
3472,Additional Mathematics
3754,Entrepreneurial Studies
3766,Business
3756,Principles of Accounting
3767,Economics
4531,Physics
4541,Chemistry
4551,Biology
6531,Chinese Language
```

The list above is in the format of SubjectCode, SubjectName separated by comma `,`. Copy and paste the list above in `AcademicSubjects.txt` file, meaning that the content of that file should look **EXACTLY** as the content shown above.

Your task for this page will be to display the subjects enrolled by the user in ascending order based on the subject name. Thus, you use the file content above to map the academic subject codes to their relevant subject names. The result should look similar to the following:

Example Result for student s100201

Enrolled Subjects:

```
=====
3472: Additional Mathematics
4551: Biology
4541: Chemistry
6531: Chinese Language
1119: English Language
1249: History
1103: Malay Language
1449: Mathematics
1225: Moral Education
4531: Physics
```

Note that you will have to write the logic for sorting instead of hardcoding it, i.e. ordering it manually and write into an array is not accepted.

Co-curriculum Page [1 mark]

As described in Project Introduction, this page shows the co-curriculum activities the student is involved in. For context, students in Malaysia are required to participate in **ONE** club for each activity category shown below:

- Societies
- Uniform Body
- Sports Club

Below I have provided a sample list of clubs from [Chung Ling High School](#):

ClubSocieties.txt
B01, Scout B03, Police Cadet B07, The Boys' Brigade P27, Computer Society P81, Young Entrepreneur Society P82, Robotic Club S01, Badminton Club S10, Swimming Club S15, Fencing Club

The format of the list above is Club Code, Club Names separated by comma ' , '. The codes starting with P refers to Societies, B refers to Uniform Body, S refers to Sports Club. Copy and paste the list above in **ClubSocieties.txt** file, meaning that the content of that file should look **EXACTLY** as the content shown above.

Similar to the Academic Page, your task for this page will be to display the clubs joined by the user, in the order of Society, Uniform Body and then Sports. Thus, you use the content in **ClubSocieties.txt** file to map the club codes to their relevant club names. The result should look similar to the following:

Example Result for student s100201
Your Cocurricular Clubs: ===== Societies: P82 - Robotic Club Uniform Body: B07 - The Boys' Brigade Sports Club: S01 - Badminton Club ===== Generate Transcript?

Note that there is a “Generate Transcript?” at the bottom of the output, this link us to the next page where the students can display their co-curriculum marks. More about it at the next section.

Co-curriculum Marks Calculator [3 marks]

This task focuses solely on the calculation system for the student’s Co-curricular activities which then generates transcript containing the student’s final Co-curriculum mark. The calculation scheme for each club is based on the following table:

Extra Simplified Calculation Scheme	
Item	Maximum Marks
Attendance	50
Position in Club	10
Level of Activities Participated	20
Achievement Level	20

For simplification, we assume every student get full attendance thus full 50 marks for each club.

The other scoring scheme is displayed below:

Position in Club Calculation Scheme (Extra Simplified Version)	
Position	Awarded Marks
President	10

Vice President / Secretary / Treasurer	9
Vice Secretary / Vice Treasurer	8
Committee	7
Active Member	6

Level of Activities Participated Calculation Scheme (Extra Simplified Version)	
Level of Activities Participated	Awarded Marks
International	20
National	15
State	12
School	10

Achievement Level Calculation Scheme (Extra Simplified Version)	
Achievement Level	Awarded Marks
Gold	20
Silver	19
Bronze	18
Participation	0

To get things started, we have the School-Level Activities Log where students can submit the activities they participated to gain merit marks:

ActivitiesLog.txt
s100201,P82,National Robotics Competition 2024,National,Gold s100202,S15,Piala Warisan 2024,National,Bronze s100201,S01,Badminton Internal Tournament 2024,School,Participation s100202,P27,CLHS Science Symposium 2024,School,Participation s100202,B01,King's Scout Pre-Rating Test Camp,National,Participation

The format of the list above is Student Matric Number, Club Code, Activity Name, Activity Level, Achievement, separated by comma ' , '. To understand it further, line 1 will be "Student s100201 got Gold Award in World Robotics Olympiad 2024, an International competition, for the Club P82 - Robotics Club". Copy and paste the list above in **ActivitiesLog.txt** file, meaning that the content of that file should look **EXACTLY** as the content shown above. Feel free to add more if you want.

To reduce unnecessary complexities to achieve this basic feature, there will only be maximum 1 activity per club for this basic feature. You may look at the Additional Feature section where students can have more than 1 activity participated for that club, tackling this challenge will be counted as additional feature mark.

Also, we have the School-Level Positions Document that stores the positions each student holds for each club:

StudentPositions.txt
s100201,Active Member,President,Vice Secretary s100202,Treasurer,Committee,Vice President

The format of the list above is Student Matric Number, Position for Society, Position for Uniform Body, Position for Sports Club, separated by comma ' , '. To understand it further, line 1 will be "Student s100201 is Active Member for P82 - Robotic Club, President for B07 - The Boys' Brigade and Vice Secretary for S01 - Badminton Club". Copy and paste the list above in **StudentPositions.txt** file, meaning that the content of that file should look **EXACTLY** as the content shown above.

If you created a new student in **UserData.txt** file, please add a new line giving the new student 3 mock positions.

Here are some cases when performing the calculation, which applies to all 3 clubs for each student.

Case 1:

The student only participated in 1 external activity for that club, the marks will be based on that club only..

Example Calculation Result for student s100201	
[S01 - Badminton Club]	
Attendance: assume full ----->	50/50 marks
Position: Vice Secretary ----->	8/10 marks
Selected Activity: Badminton Internal Tournament 2024	
Level of Activities: School ----->	10/20 marks
Achievement Level: Participation ->	0/20 marks
=====	
TOTAL: 68/100 marks	
=====	

Case 2:

The student did not participate in any external activity for that club, thus no marks is awarded for the activity section.

Example Calculation Result for student s100201	
[B07 - The Boys' Brigade]	
Attendance: assume full ----->	50/50 marks
Position: President ----->	10/10 marks
Selected Activity: None	
Level of Activities: None ----->	0/20 marks
Achievement Level: None ----->	0/20 marks
=====	
TOTAL: 60/100 marks	
=====	

We can generate the transcript by repeating the calculation for all 3 clubs for that student, then the final marks for the student's co-curriculum activity will be the average of top 2 clubs with highest marks. Let's take a look at the transcript below:

Transcript for student s100201

Co-curriculum Transcript for s100201

[P82 - Robotic Club]

Attendance: assume full -----> 50/50 marks

Position: Active Member -----> 6/10 marks

Selected Activity: National Robotics Competition 2024

Level of Activities: National -----> 15/20 marks

Achievement Level: Gold -----> 20/20 marks

TOTAL: 91/100 marks

[B07 - The Boys' Brigade]

Attendance: assume full -----> 50/50 marks

Position: President -----> 10/10 marks

Selected Activity: None

Level of Activities: None -----> 0/20 marks

Achievement Level: None -----> 0/20 marks

TOTAL: 60/100 marks

[S01 - Badminton Club]

Attendance: assume full -----> 50/50 marks

Position: Vice Secretary -----> 8/10 marks

Selected Activity: Badminton Internal Tournament 2024

Level of Activities: School -----> 10/20 marks

Achievement Level: Participation ---> 0/20 marks

TOTAL: 68/100 marks

FINAL MARKS: 79.5 marks

The final marks is calculated by taking the 2 highest divided by 2: $(91 + 68) / 2$

Suggested Extra Features (4 Marks)

Marks at this section are given based on the amount of impact or significance that the suggested features or extra features by students has towards the project as a whole. It is SUBJECTED to the demonstrator or the lecturer giving the marks to decide how much marks should be awarded for each extra features.

Graphical User Interface

A graphical user interface (GUI) is a digital interface in which a user interacts with graphical components such as icons, buttons, and menus. In a GUI, the visuals displayed in the user interface convey information relevant to the user, as well as actions that they can take. A nice-looking and user-friendly GUI will give the user a better experience using the software. You may choose to use [JavaFX](#) or [Spring Boot](#) with other technologies to do so.

Relational Database

A relational database is a collection of information that organizes data in predefined relationships where data is stored in one or more tables (or "relations") of columns and rows, making it easy to see and understand how different data structures relate to each other. Relationships are a logical connection between different tables, established on the basis of interaction among these tables. You may use [Oracle Database](#), [MySQL](#), [Firestore](#) or other relational databases to do so.

Note that the parts where you are required to copy and paste the content in this assignment question into a txt file like the **AcademicSubjects.txt**, you may load them into your database instead of writing it in a txt file. Just to remind that you are still required to have **AT LEAST ONE** file I/O application in order to get the mark for Data Storage at Basic Feature.

Password Hashing

Storing the raw passwords of our users in any kind of storage violates the basic privacy of the users. According to the [United Kingdom's Article 5 of Regulation \(EU\) 2016/679](#) of the European Parliament and of the Council, personal data shall be processed in a manner that ensures appropriate security of the personal data, including protection against unauthorised or unlawful processing and against accidental loss, destruction or damage, using appropriate technical or organisational measures ('integrity and confidentiality'). You may use hashing, Caesar cypher, or any other encryption

algorithm to tackle this matter. However, you must be able to justify the algorithm during the presentation. Plus, you should also show that your database or the text file's password section stores the hashed version of the password.

Emailing

You may set up emailing system to be used anywhere within the project. An example use case it to give the user an option to email them their co-curriculum transcript as a PDF format.

Handling Selection From Multiple Activities

Going back to the Calculation section, the new challenge here is if the student participated in more than 1 external activity for that club. You may append the following into the **ActivitiesLog.txt** file

ActivitiesLog.txt
s100201,P82,World Robotics Olympiad 2024,International,Silver s100201,P82,FIRST Lego League 2024,National,Gold

For this case, the marks will be taken on the activity with the highest marks.

Example Calculation Result for student s100201
[P82 - Robotic Club] Attendance: assume full -----> 50/50 marks Position: Active Member -----> 6/10 marks Selected Activity: World Robotics Olympiad 2024 Level of Activities: International -> 20/20 marks Achievement Level: Gold -----> 19/20 marks =====
TOTAL: 95/100 marks =====

Expanding Functionalities on Existing Features

Since this project is super basic compared to existing systems like UM's academic portal [Spectrum](#) or student portal [Mysis](#), you are highly encouraged to add some functionalities to each pages or existing features. One example is to include all

activities participated in the transcript rather than just showing the selected ones.

Note that the impact of your added functionalities affect the marks awarded as extra feature, i.e. more impactful or significant functionality rather than a small add-on grants you more marks.

Tips For This Assignment

To help you complete the assignment easily, here are some tips from our experiences in doing project-based assignments.

Modularity

This project is modular and can be separated into a few parts:

- Login / User Registration
- User Class creation and Data Storage
- Academic Page
- Co-Curriculum Page
- Co-Curricular Activities Marks Calculation

This eases your team to delegate the tasks among the team members to effectively construct each functionality separately and compile together after testings have been done. Note that the separation above is just a suggestion and not compulsory to follow while your team distributes tasks and responsibilities.

Version Control

We encourage you to utilize Git versioning and the GitHub platform while collaborating with your team to complete this project. You may read more about Git and GitHub in the following links:

- [What is Git?](#)
- [Getting Started on GitHub](#)
- [Creating Pull Requests on GitHub](#)
- [Resolving Merge Conflicts on GitHub](#)

Relative Path

Since there are many File I/O involved in this assignment, there is a tendency that developers used their local absolute file path in their code. Here's a sample difference between absolute path and relative path:

- Absolute Path:
C:\Users\Documents\WIX1002\Assignment\SampleInput.txt
- Relative Path:
./SampleInput.txt (given that the project root ~ is at the \Assignment\ folder as shown in the Absolute Path)

We strongly advise you to NOT use the absolute path but use the relative path since the file path in your PC will NOT necessarily be the same for your other project collaborators, but the [file hierarchy](#) in the project should be the same for all of your project collaborators.

Contact Me

If you have any questions or need clarifications about the assignment, please contact me, Lim Jun Yi, using either of the following methods:

- WhatsApp me at [+60123681620](https://wa.me/60123681620)
- Email me at 22004811@siswa.um.edu.my

I will try my best to answer your questions as soon as possible. Hope you enjoy this assignment!