**PRIVATE ACCOMMODATION RENT SYSTEM**

The Asia Pacific Home (APH) is intended to design and develop a rent system for Klang valley, Malaysia. The system will primarily publish or post the advertisements from the interfacing system – Advert web (mudah) - by importing adverts listing, and enable the renting functionality including report generation to the clients.

The APU tech-solution has been successfully tendered the project with the solution proposal and granted to work on for further detailed requirements, technically. In this case, you and your other **THREE (3)** team members should use the prepared *mudah-apartment-kl-selangor.csv* dataset to support the system. This dataset contains information of more than 10,000 property with rental pricing surrounding Kuala Lumpur and Selangor, Malaysia. It also includes the following attributes:

* ads\_id: the listing ids (unique)
* prop\_name: name of the building/ property
* completion\_year: completion/ established year of the property
* monthly\_rent: monthly rent in ringgit malaysia (RM)
* location: property location in Kuala Lumpur region
* property\_type:property type such as apartment, condominium, flat, duplex, studio, etc
* rooms: number of rooms in the unit
* parking: number of parking space for the unit
* bathroom: number of bathrooms in the unit
* size: total area of the unit in square feet
* furnished: furnishing status of the unit (fully, partial, non-furnished)
* facilities: main facilities available
* additional\_facilities: additional facilities (proximity to attraction area, mall, school, shopping, railways, etc)

Therefore, your team is required to implement the prototype through C++ programming and implement appropriate data structures to store and manage the property details, users’ personal details and etc. As required by APU tech-solution, the system capabilities must include, **but not limited to**, the following options:

**Tenant (tenant)**

1. 1. Login/register and logout.
   2. Sort and display property information in descending order based on monthly rent, location, and size as per square feet.

* *Apply 2 distinct sort algorithms in this function to compare their execution time.*
* *Discuss the result in the final report as shown in Appendix 2*
  1. Search and display property details based on tenant/customer decision.
* *Apply 2 distinct search algorithms in this function to compare their execution time.*
* *Discuss the result in the final report as shown in Appendix 2*
  1. Save their favorite property.
  2. Place a rent request the desired property that is stored in the favorite list.
  3. Display property renting history.

**Manager**

* 1. Login and logout.
  2. Display all registered tenant’ details.
  3. Search tenant’s details.
  4. Delete tenant accounts based on inactivity status.
  5. Based on the list of all favorite property list, summarize the top 10 property, and generate a report.
  6. Manage tenancy process upon renting request received.
  7. Manage payment of the confirmed tenancy.

**Admin**

* 1. Login and Logout
  2. Add new manager user and modify the status (**active**: able to login; **inactive**: unable to login)
  3. Display all tenants and property information with filtering criteria.

**Additional information for this assignment:**

1. You need to include **TWO (2)** distinct searching and **TWO (2)** sorting algorithms in the system. Evaluate the algorithms’ efficiency. Justify your choices and explain in detail how they can help you to make your system more efficient.
2. This assignment does not require any SQL or NO-SQL database to store any details.
3. Include all validations required for the system and use good programming practices (e.g., indentation, meaningful identifier names, comments, etc.).

**Assignment Requirements**

A group can **only have a maximum of 2-3 students.**

There are two submissions of the required in this assignment:

1. **Technical Proposal Stage (Week 8)**
   * You need to submit your technical proposal on the **LMS in week 8**.
   * Proposals should include proposed data structures, overall system workflow, sorting and searching algorithms to be used in the system.
   * These sorting and searching algorithms should be presented as **flowcharts or pseudocode.** Compare them on a theoretical basis and justify why you use them.
   * This proposal will contribute **20%** for the final assignment marks (under Design criteria).
   * Refer to the Appendix 1 for the PowerPoint Slides contents.

***Approximation of Total Pages for the PowerPoint slides: 12 - 15 (max).***

1. **Final Report and Presentation Stage (***follow the hand-in date***)**

You are required to submit a **softcopy** of the assignment report and source code to the Moodle system. The report should contain:

* Detailed explanation of the data structures and classes created, with proper justification on your decisions (include source code defining classes, data members, and method headers only).
* Brief explanation about the algorithms used to implement the functionalities stated above (include code snippets of important parts of implementation).
* Source code of the main function, with screenshots showing program’s input and output interactions.

*Approximation of Total Pages for the documentation: 30 - 60 (max).*

*Approximation of Words for the documentation: 2000 words (min)*

You must **present your assignment solution and answers** to the lecturer during a Q&A session that will be conducted after the hand-in date.

If you use some code which has been taken or adapted from another source (book, magazine, internet, forum, etc.) then this must be **cited and referenced** using **APA Referencing Style within your source code**, and this must be mentioned explicitly in the **report**. Failure to reference code properly will be treated as plagiarism. **Automated tools for checking code similarities** among submissions will be used, and all detected cases will be treated as cheating.

Assessment marks are divided as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Total Percentage** | **80%** | | **20%** | |
| **Design (proposal)** | **Implementation** | **Documentation** | **Presentation** |
| **Total raw marks in each section** | 25 | 75 | 50 | 50 |

**What You Need to Hand In during the final submission?**

1. You are required to hand in the group project report on or before the due date mentioned on the cover sheet of the assignment.
2. A softcopy of the report (in Word Document / PDF format), in addition to the C++ files of the programs. The organization of files and folders must adhere to the following instructions precisely:

* The report should be named using format

*“<GroupNo>\_<student ID-leader>\_<student ID-member1>\_<student ID-member2>\_<student ID-member3>.docx”.*

For example **“G1\_TP012345\_TP012344\_TP012123\_TP012126.docx”**

* All the source codes **(.cpp and .h)** should be zipped into one file and named following the above format. Make sure to **DELETE** all non-source-code files, including executables (\*.exe).

1. You should **present an executable solution** during Q&A session to demonstrate program execution, the working of the data structure, your understanding of the code, and ability to modify / fix it.

**Marking Criteria:**

The program submitted will be evaluated according to the following performance criteria:

**Distinction (90% and above)**

* Program compiles and executes perfectly
* At least 90% of the required functionalities are correctly implemented
* Efficient data structures and\or algorithms are used in the implementation
* Clear coding style and structure, and code is properly commented
* Functionalities are fully tested/validated in program execution

**Credit (70% – 89%)**

* Program compiles and executes
* Between 70% and 90% of the required functionalities are correctly implemented
* Implementation uses a data structure or algorithm that is not most efficient
* Clear coding style, and code is properly commented
* Functionalities are not fully tested/validated in program execution

**Pass (50% - 69%)**

* Program compiles perfectly and executes
* Between 50% and 70% of the required functionalities are correctly implemented
* Implementation uses inefficient data structures or algorithms
* Unclear coding style, or code is not properly commented
* Functionalities are not full tested/validated in program execution, or produce errors in some cases

**Marginal Fail (30% - 49%)**

* Program does not compile or run, but coding logic is almost correct
* Between 30% and 50% of the required functionalities are correctly implemented
* Implementation uses inefficient data structures or algorithms
* Unclear coding style, and no comments provided
* Functionalities are not tested/validated in program execution

**Fail (below 30%)**

* Program is not given
* Program does not compile or run
* Less than 30% of the required functionalities are implemented
* Implementation uses very inefficient data structures or algorithms
* No proper code structure and no comments provided

**APPENDIX 1: SHORT PROPOSAL OF YOUR GROUP PROJECT:**

*Note: Approximation of Total Pages in PowerPoint Slides: 12 - 15 (max).*

Before your group start to develop the system, your group is required to write a short proposal to identify what are the data members will be included your structure, what are workflows involved in the system, and which of the algorithms will be implemented in your future system.

Below are the minimum elements that you need to include in your slides:

1. Introduce your group’s members (*1 slide*).
2. Identify what are the important data structures or classes (optional) will be included in your system (*2 – 4 slides*).

* Visualization of data structures
* A detailed description of the data structures and classes created, with appropriate justification for your decision (including only source code defining classes, data members, and method headers).

1. Briefly explain all the workflows planning for your future system. (*5 – 6 slides*)

* Use flowchart / pseudocode to describe whole system workflows. No need to explain every functional workflow.
* Provide a brief explanation about the **TWO (2)** search and **TWO (2)** sort algorithms that you plan to use in your system.
* Compare all the search and sort algorithms.
* Provide proper justification for your decisions.

1. Provide a workload distribution table to determine the tasks of the members. (*1 slide*)

You are required to submit a softcopy of the proposal to the LMS before week 8. If your group fails to submit the proposal report, it will cause your group to get a 0 mark from your design section (20%).

Assessment marks in design section are divided as follows:

|  |  |  |
| --- | --- | --- |
| **Total Percentage** | **Design Section (20%)** | |
| **Data Structure (design)** | **System Workflow and Algorithm** |
| **Total raw marks in each section** | 10 Marks | 15 Marks |

**APPENDIX 2: OUTLINE OF YOUR FINAL REPORT:**

The final report outline as below:

* Cover Page
* Module Code and Name
* Intake code
* Proposal Title
* Include your group member list.
* Introduction
* Brief introduction about the system
* **Source code of each developed data structure / class**
* Detailed explanation of the data structures and classes created.
* Implementation and results:
* **Source code of each algorithm**.
* System Input / Output Screenshot
* Brief explanation about all screenshots
* Compare the execution time between **TWO (2)** selected search algorithms and discuss the results.
* Compare the execution times of **TWO (2)** selected sort algorithms and discuss the results.
* Conclusion, Future Works, and Reflection.
* A summary about your system (including limitations)
* Brief description of your future work based on your system limitations.
* Briefly describe your experience/feedback on the assignment
* References
* Appendix: Workload Matrix Table with signature