# Bachelor of Computer Science (Hons) Bachelor of Software Engineering (Hons)

**Module Code:** ITS66704 (July/Aug 2023)

**Module Name:** Advanced Programming

|  |  |  |  |
| --- | --- | --- | --- |
| **Assignment No./Title** | **Assignment Task 2 & Task 3 (Group Project)20% (PART A - ANALYSIS AND DESIGN)**  **30% (PART B - DEVELOPMENT)**  **10% (PRESENTATION)** | | |
| **Course Tutor/Lecturer** | Mr. Biju Maharjan/ Mr. Subit Timalsina | | |
| **Submission Date** | 09/11/2023 (PART A - ANALYSIS AND DESIGN)  23/11/2023 (PART B - DEVELOPMENT)  23/11/2023 (PRESENTATION) | | |
| **Student Name, ID and Signature** | | | |
| **Student Name** | | **ID** | **Signature** |
| Pragati Kumar chaudhary | | 0355530 |  |
| Sijan shah | | 0355471 |  |
| Swornim Shrestha | | 0355615 |  |

**Declaration** (need to be signed by students. Otherwise, the assessment will not be evaluated)

Certify that this assignment is entirely my own work, except where I have given fully documented references to the work of others, and that the material contained in this assignment has not previously been submitted for assessment in any other formal course of study.

|  |  |
| --- | --- |
| **Marks/Grade:** | **Evaluated by:** |
| **Evaluator's Comments:** | |

Table of Contents

1. Introduction1
2. Application description and rational 2
3. Role and responsibility of each group member3
4. Testing plan and implementation5
5. UML Use case and UML class diagrams and Screenshot of the Output of the program6
6. References………………………………………………………………………………………………….7

**Acknowledgement**:

First, we would like to express our gratitude to Taylor's University and our lecturers, Mr. Biju Maharjan, and Mr. Subit Timilsina, for providing us with helpful guidelines for our assignment during numerous.

consultations. We would like to express our gratitude for your guidance and assistance in

helping us complete this assignment. We want to express our gratitude to him for giving us.

the data and materials we needed for this project, as well as to the university for letting us.

complete this writing report assignment.

Additionally, our group members' efforts and cooperation are necessary for us to finish this.

assignment. Because each member of our group effectively filled their assigned role, the

assignment was successfully completed. We always put a lot of effort into producing a

quality assignment with our devotion and responsibility.

1. **Introduction**:

An innovative effort called the Citizenship Assessment System attempts to meet the increased demand for cultural integration and civic education among recently naturalized citizens. This paper presents a thorough analysis, design, and development of this novel system, which aims to equip people with information about the history and culture of their new nation. A tool such as the Citizenship Assessment System becomes more and more important in an era of rising migration, cultural diversity, and border mobility. By outlining the project's background and importance, this part establishes the scene.

In conclusion, there is a great deal of social and educational value to the Citizenship Assessment System project. It can promote cultural integration, civic consciousness, and informed citizenship among a varied and expanding community of naturalized citizens by expediting the assessment process and providing insights into the history and culture of the host nation. The study, design, and development of this system will be covered in detail in this article, along with information on its parts, features, and the Object-Oriented Programming (OOP) concepts that form its framework.

1. **Application Description and Rationale**

The purpose of the Citizenship Assessment System is to provide a quick and easy way for newly naturalized citizens to integrate into their host countries. A series of well written questions on the histories and cultures of the individual nations are available on this web-based platform. Users can test their knowledge and learn more about the country they currently call home by taking these assessments. A wide range of users, including those from different age groups and backgrounds, may access the system thanks to its user-friendly design**.**

Main Features that the system include:

* **Question Bank:** Twenty questions covering historical, cultural, and geographic characteristics are compiled for each chosen nation.
* **User Profiles:** A user's profile can be created, evaluation history can be monitored, and progress can be measured. Means any user who wants to play the quiz game can register first and then enjoy the quiz game.
* **Scoring and Feedback:** Users can pinpoint areas for improvement with the aid of prompt scoring and thorough comments following each evaluation.
  1. **Rationale**

The Citizenship Assessment System is being developed for several very good reasons**.**

**Cultural Integration:** It is essential to encourage cultural integration as countries grow increasingly multicultural and diverse. A sense of harmony and togetherness can be established across various communities by fostering a knowledge of a country's history, culture, and values.

**User-Friendly Design:** The system's intuitive user interface prioritizes inclusion in its design. People of different ages and backgrounds, even those with no computer proficiency, can use it. This guarantees that a wide range of users can take advantage of the system.

**Educational Support:** In addition to evaluation, the system functions as an educational support tool by giving users instant feedback and connections to further resources. It helps users get the knowledge they need to succeed in their new community and leads them toward a deeper understanding of their new country.

In conclusion, the goal of the Citizenship Assessment System is to produce knowledgeable and involved citizens by expediting the evaluation procedure and providing information about the history and culture of their new nation.

1. **Role and responsibility of each group member**

|  |  |
| --- | --- |
| Group member name | Roles and responsibility |
| Pragati Kumar chaudhary | Backend functions, Testing |
| Sijan shah | UI UX design, Report writing |
| Swornim Shrestha | Report documentation |

**Sijan shah**

Role: UI/UX Development

Responsibilities:

* The Citizenship Assessment System's user interface is being designed and developed using JavaFX.
* Using JavaFX, design and develop the system's front-end features, such as user registration, assessments, results display, and navigation.
* Work together with Pragati Kumar Chaudhary, the backend developer, to incorporate the JavaFX UI with the backend features.
* To enhance the user experience, test the JavaFX-based user interface with real users and collect their input.

**Pragati Kumar chaudhary**

Role: Backend App Functions

Responsibilities:

* Using Java, create and implement the Citizenship Assessment System's back-end features.
* Work together with Sijan Shah, the front-end developer, to incorporate the back-end features into the JavaFX-based user interface.
* To protect user data and system operations, implement strong user authentication procedures.
* To guarantee dependability and security, thoroughly test database functions and back-end components.

**Swornim Shrestha**

Role: Report Writing and Documentation

Responsibilities:

* Write a comprehensive, logically organized report that details every aspect of the project, including Parts A and B.
* Assemble and arrange project-related data, including the project's description, analysis, design, and development specifications.
* Work together with other team members to compile the data and updates needed for the report.
* Make that the report is formatted correctly and has an introduction and justification, along with all pertinent information.

1. Testing plans and implementation

**4.1 Testing plan:**

To make sure the Citizenship Assessment System runs consistently, safely, and according to plan, a thorough testing strategy is essential. A variety of testing methods are included in the plan to confirm the security, performance, and operation of the system.

Types of Testing

1. **Unit Testing:** Verify that each of the application's parts, features, and operations operates as intended. This entails testing features like database operations, assessment scoring, and user registration.
2. **Integration Testing:** Examine the communication between the JavaFX front-end and Java back-end components. Check to make sure the system runs smoothly, and that data is sent accurately.
3. **User Interface (UI) Testing:** Assess the usability and responsiveness of the user interface. Make sure users can engage with the system easily and that it is intuitive**.**
4. **User Experience (UX) Testing:** Evaluate the entire user experience, considering the application's flow, navigation, and user comments. Ensure that the system has a pleasing appearance and is easy to use.
5. **Database Testing:** Check the security and correctness of the data retrieval and storage. To guarantee that user data is protected, test user authentication systems.

**4.2 Testing implementation**

**Integration Testing:** Create test cases that address the points of integration between the Java backend and the JavaFX front end. Make certain that the data is sent correctly.

**User Interface (UI) Testing:** Verify the user interface's responsiveness by manually testing it across a range of devices and browsers. Make sure all the components are functional and positioned correctly.

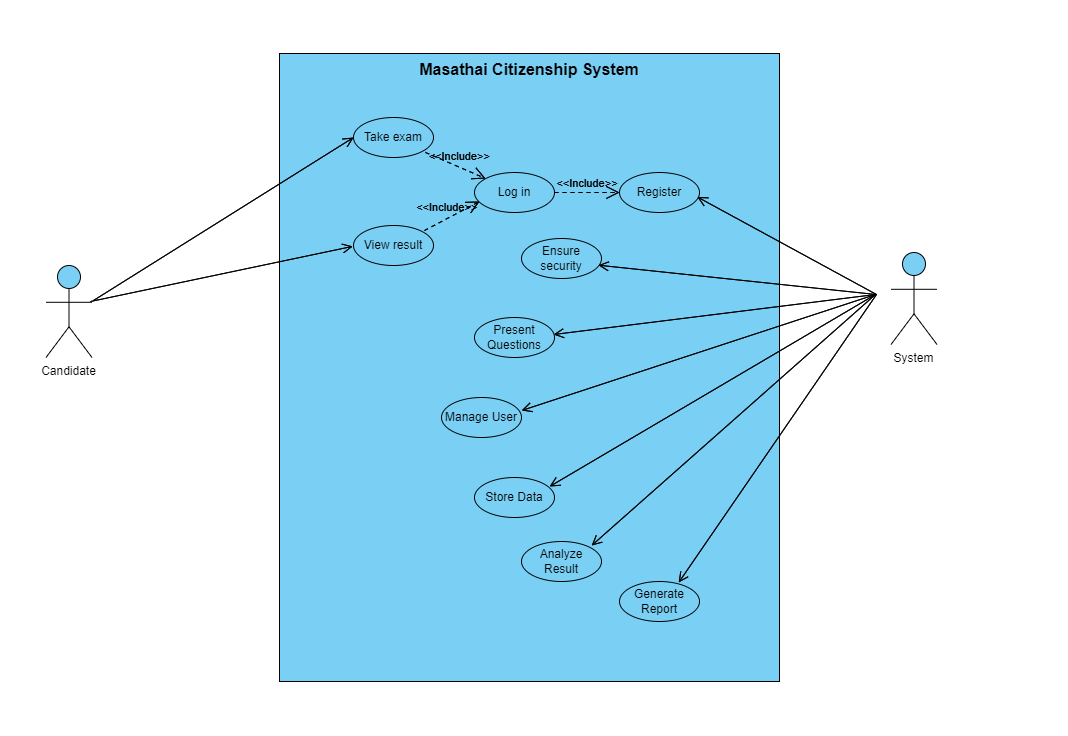
**User Experience (UX) Testing:**

Allow actual users or interested parties to explore the application. After gathering user experience input, make the required adjustments.

**Database Testing:**

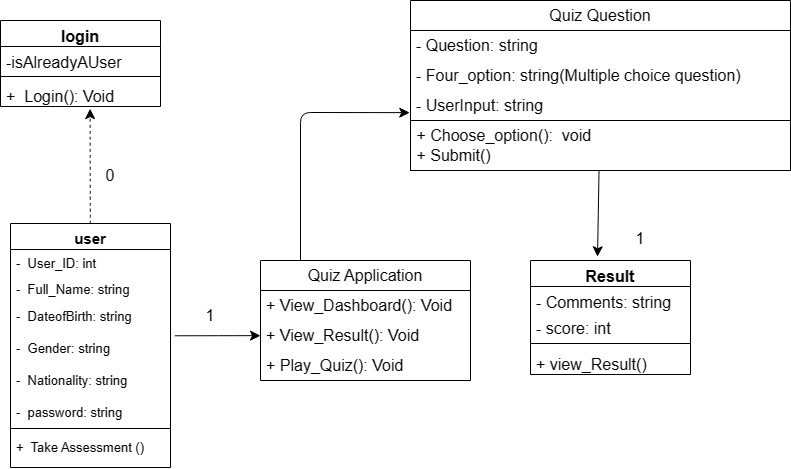
To ensure that data is appropriately saved and retrieved from the MySQL database, use SQL queries. Verify user identity and access restrictions.

**UML case and Use case UML Diagram**

****

**Figure 1 UML Use case diagram for the Masathai Automated System**

2.) UML Class Diagram for the Masathai Automated System:



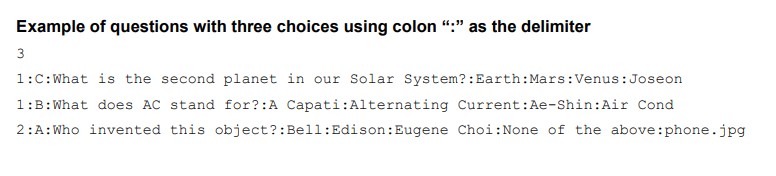
**Screenshot of the output of the program**

# References

Anon., 2023. [Online]   
Available at: https://www.javaguides.net/2019/07/registration-form-using-java-swing-jdbc-mysql-example-tutorial.html

Anon., 2023. *Blogspot.* [Online]   
Available at: https://1bestcsharp.blogspot.com/2018/05/java-login-and-register-form-with-mysql-database.html

Anon., 2023. *java-login-form-Registerform.* [Online]   
Available at: https://github.com/Anas436/Java-Login-Form-Register-Form-with-MYSQL-Connectivity



# APPENDIX A

**MARKING SCHEME – PART A**

|  |  |
| --- | --- |
| **Items** | **Marks** |
| Works responsibilities and delegation | 3 |
| Chosen countries brief | 1 |
| Minimum of 20 potential questions | 5 |
| UML Use case diagram | 2 |
| UML class diagram | 4 |
| User Interface static prototype | 4 |
| Overall documentation | 1 |
| **TOTAL** | **20** |

**MARKING SCHEME – PART B**

|  |  |
| --- | --- |
| Existence of Type A question | 1 |
| Existence of Type B question | 1 |
| Existence of Type C question | 2 |
| Operational Candidate Form – verify id, details entry and proceed button | 3 |
| Operational Examination Form – name, gender and flag | 1 |
| Test Form question number status and navigation | 1 |
| Countdown timer | 2 |
| Proper candidate.txt input file | 1 |
| Proper question\_list.txt input file | 1 |
| Proper test\_result.txt output file | 1 |
| Operational Analysis Form and sorted list | 2 |
| Operational Result Form - dropdown list and correct answers | 2 |
| Proper variable name, indentation & comments | 2 |
| Source code and screenshot of the system | 5 |
| Proper submission file organization and documentation | 2 |
| Aesthetic | 3 |

# TOTAL 30

**Bonus (max 2 marks)**

# Items Marks

BONUS – Serialized input data **2**

**Individual – Presentation (10 marks)**

|  |  |
| --- | --- |
| **Items** | **Marks** |
| Professionalism | 2 |
| System demonstration | 4 |
| Delivery | 2 |
| Confidence | 2 |
| **TOTAL** | **10** |

# MARKING RUBRICS

**For EACH criterion of marks allocated, the following rubrics will be applied:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **100% of**  **allocated marks** | **75% of**  **allocated marks** | **50% of**  **allocated marks** | **25%**  **allocated marks** | **0%**  **allocated marks** |
| * Complete understanding of the problem * A plan that could lead to a correct solution with no algorithmic errors * Correct solution | * Misinterprets minor part of the problem * Substantially correct solution with minor omission or procedural error | * Misinterprets major part of the problem * Partially correct solution but with major fault * Computational error, partial solution for problem. | * Completely misinterprets the problem * Substantially inappropriate solution | * No attempt * No answer or wrong answer based upon an inappropriate solution |