CHAPTER 4

SYSTEM DESIGN AND IMPLEMENTATION

4.0 Overview

This chapter describes design and implementation of a patient simulation learning. The system structure will be described in several parts which are the system flowchart, the system interface, use case diagram and sequence diagram. Figure 4.1 Illustrate the flowchart of the patient simulation for eye disability diagnosis.

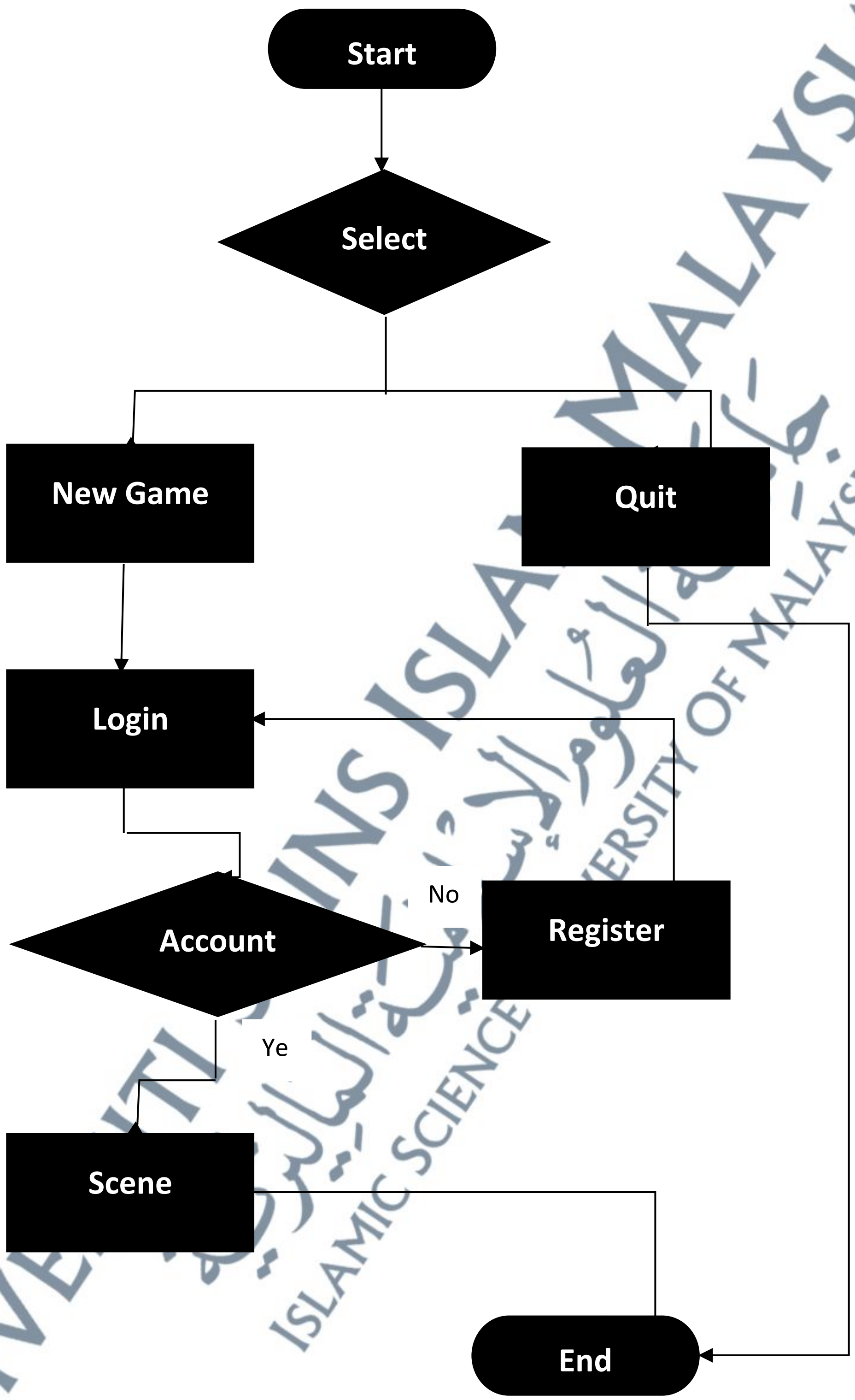

Figure 4. 1 Interface Flowchart

4.1 Use Case Diagram

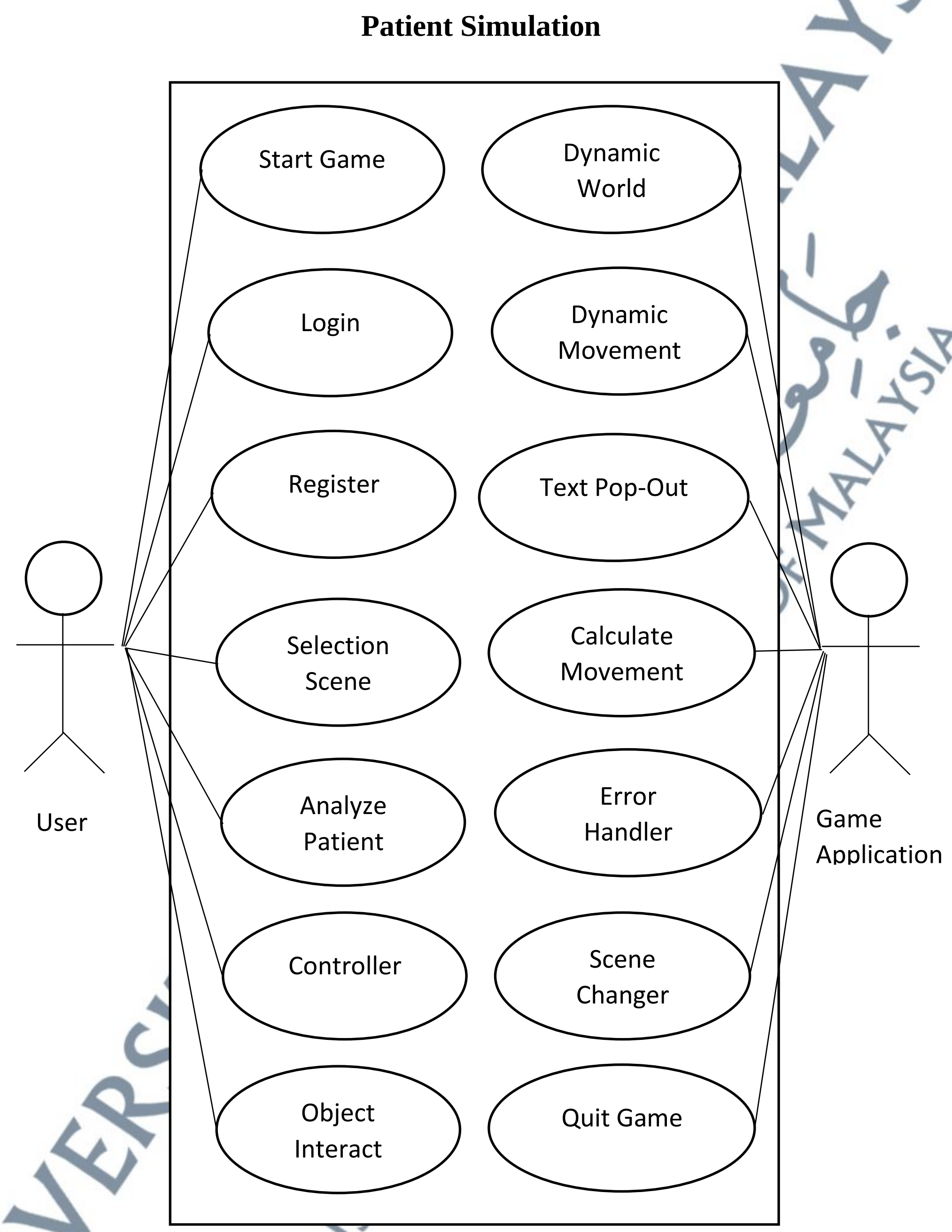

Figure 4. 2 Use Case Diagram

Figure 4.2 illustrates the use case diagram specific to the patient simulator software, an integral component of game development. The user interface (UI) assumes a pivotal role as it serves as the primary means of interaction between players and the game. A well-designed UI substantially enhances the player's experience by ensuring the game is intuitive and easy to navigate. Conversely, a poorly designed UI can lead to frustration and hindered gameplay. Within the patient simulator software, the UI encompasses essential elements such as the start game interface, login interface, register interface, and score display. These elements furnish players with crucial information and enable interaction through buttons or controls that facilitate movement, interactions, and other in-game features. Additionally, the UI incorporates menus, buttons, and icons to facilitate access to different sections, such as the options menu or the login system.

Effective UI design prioritizes intuitiveness and ease of understanding, enabling players to quickly grasp the game mechanics. Visual appeal and consistency assume equal significance as they aid players in identifying important elements and comprehending the game's structure effortlessly. In the context of patient simulation, the design of UI elements is informed by user experience surveys to ensure they align with the expectations and requirements of the target audience. Moreover, accessibility plays a crucial role in UI design, necessitating game developers to consider players with disabilities and incorporate features such as larger text, alternative color schemes, and adaptable control methods.

Patient simulation stands as a vital application within the realm of game development, encompassing critical features such as cursor movement calculations and error handling through text displays. Prior to embarking on new game development projects, conducting an extensive study of existing game applications confers several benefits. Such an analysis grants developers insight into current market trends and popular gaming features, enabling them to identify opportunities for innovation and differentiation. By leveraging these insights, developers can create games that stand out in the market and resonate with a specific niche of players.

In summary, studying existing game applications represents an indispensable step in the game development process, yielding valuable insights into the market, competition, and overall industry landscape. Armed with this knowledge, game developers are empowered to craft engaging and successful games. Each game application requires user actions or interactions with specific scene elements to activate system functionalities. For instance, initiating the patient simulation game requires users to press the start button, redirecting them to the login menu. Likewise, within the simulation world interaction system, players must maneuver their cursor to specific regions of the programmed UI to trigger desired functions. To gain a more comprehensive understanding of the intricate interactions between user functionalities and the game application, sequence diagrams are created, offering an insightful overview of the system's behavior.

4.2 Sequence Diagram

Sequence diagram are interaction diagrams which describe how operations are performed. In the sense of a partnership, they catch the connection between objects. Sequence diagram are the time emphasis that physically demonstrate the order of the contact with the vertical axis of the diagram to di time as to which messages are received and when. The diagram below shows the sequence diagram for patient simulation system.

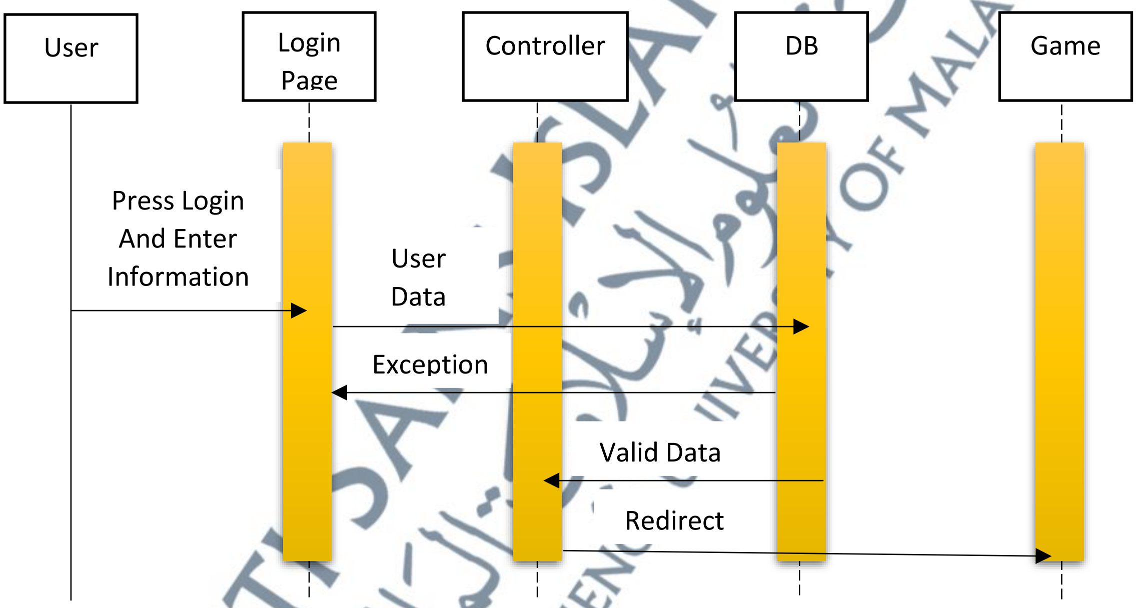

Figure 4. 3 Login Sequence Diagram

Figure 4.3 illustrates the sequence diagram specifically tailored for the login system within the patient simulator software. As an integral part of the software's functionality, the login system enables users to access the patient simulation environment by providing their credentials. The server's role in managing the login process ensures secure and authorized access to the software.

To begin, users must first register their information, including personal details and account credentials, which is then stored in the software's database. When users initiate the registration process, the server receives the registration request and verifies the availability of the chosen username. If the username is already in use, the server prompts the user to select an alternative username, ensuring uniqueness for each account.

Once a unique username is chosen, the server proceeds to store the user's information in the database, encrypting sensitive data such as passwords using hashing techniques. This encryption adds an extra layer of security, safeguarding the user's account information. Additionally, a unique session ID is assigned to the user, allowing for tracking and authentication during subsequent login attempts.

When users request to log in to the patient simulator software, they provide their username and password through the login form. The server receives this login request and retrieves the corresponding credentials from the database. The server then compares the entered username and password with the stored records to determine if they match. If the provided credentials do not match, the server sends a failure response, indicating that the login attempt was unsuccessful. The user is prompted to review their credentials and try again.

Conversely, if the username and password match the stored records, the server proceeds to verify the user's session ID. By cross-referencing the session ID associated with the user's account, the server ensures that the login request originates from an authorized session. If the session ID is valid, the server grants access to the patient simulation environment, allowing the user to begin practicing their medical skills.

Throughout the entire login process, the server plays a crucial role in managing user authentication and maintaining the security of user accounts. By encrypting passwords and verifying session IDs, the server guarantees the integrity and confidentiality of user information. Clear response messages from the server guide users in their login attempts, providing feedback and aiding them in resolving any login-related issues they may encounter.

In conclusion, the login system within the patient simulator software exemplifies the importance of secure user authentication and access control. The server's role in managing user registration, verifying credentials, and validating session IDs contributes to a safe and reliable user experience within the patient simulation environment.

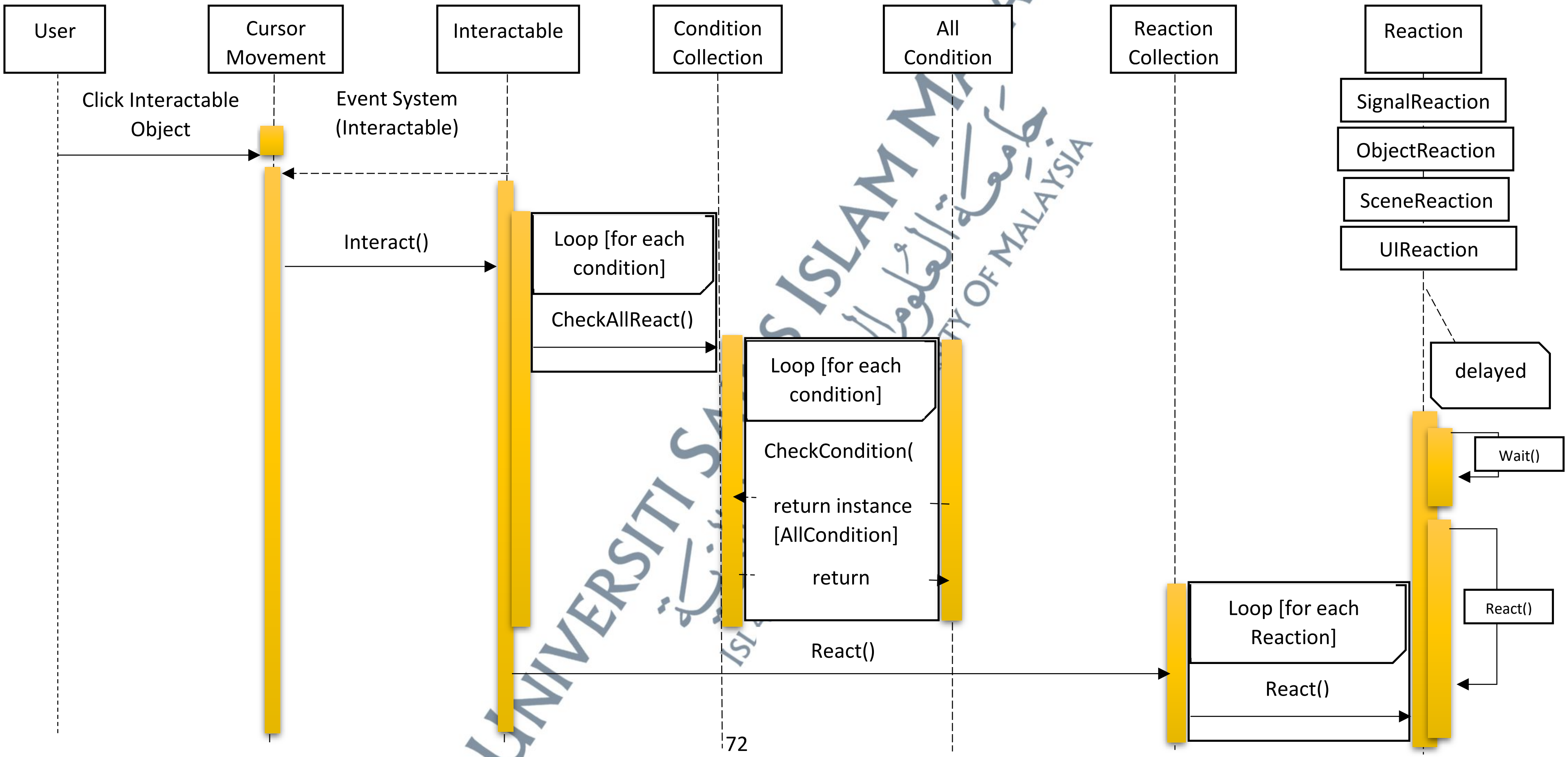
User
Cursor Movement
Interactable
Condition Collection
All Condition
Reaction Collection
Reaction
SignalReaction
ObjectReaction
SceneReaction
UIReaction
Click Interactable Object
Event System (Interactable)
Interact()
Loop [for each condition]
CheckAllReact()
Loop [for each condition]
CheckCondition(
return instance [AllCondition]
return
delayed
Wait()
React()
Loop [for each Reaction]
React()
React()

Figure 4. 4 User Click On Interactable Sequence Diagram

Figure 4.4 illustrates the sequence diagram for user interactions with interactable objects in the context of the patient simulator software developed using Unity. In the Unity framework, buttons serve as essential components of the user interface, enabling users to interact with the game and trigger specific actions. One of the commonly employed methods for button functionality in Unity involves associating them with functions that are executed upon button clicks. These functions, consisting of reusable blocks of code, enhance code organization and readability.

The OnClick event in Unity facilitates the assignment of functions to execute when buttons are clicked, simplifying the execution of specific code upon interaction. Developers can create public functions within scripts linked to buttons and assign them to the OnClick event through the Unity editor. For example, to create a button that initiates a game, developers can create a public function named "StartGame" in a script attached to the button and associate it with the OnClick event. Upon button click, the StartGame function is invoked, and the code inside it is executed to commence the game.

In conclusion, buttons play a crucial role in user interactions within the patient simulator software developed using Unity. By associating buttons with functions that execute upon clicks, developers can create reusable code, define distinct behaviors for different buttons, and enhance the interactivity and user-friendliness of the game. Leveraging the OnClick event simplifies the execution of specific code upon button press, and the ability to call functions enables the creation of more complex and engaging interactions within the software.

4.3 Simulation Algorithm

```python
# Initialize simulation parameters
patient_eye_health = 100
pen_position = [0, 0]  # Pen's initial position (x, y)
diagnosis = ""

# Simulation loop
while patient_eye_health > 0:
    # Display simulation interface with patient's eye
    display_patient_eye(patient_eye_health)

    # Get user input for pen movement (assuming arrow keys)
    user_input = get_user_input()

    # Update pen position based on user input
    if user_input == "up":
        pen_position[1] += 1  # Move pen up
    elif user_input == "down":
        pen_position[1] -= 1  # Move pen down
    elif user_input == "left":
        pen_position[0] -= 1  # Move pen left
    elif user_input == "right":
        pen_position[0] += 1  # Move pen right

    # Simulate diagnosis based on pen position and eye health
    diagnose_eye_condition(pen_position, patient_eye_health)

    # Update patient's eye health (e.g., decrease over time)
    patient_eye_health -= 1
```

1. **Initialization:**

To initiate the simulation, we start by initializing essential parameters:

- **Patient Eye Health (patient_eye_health):** This variable represents the health of the patient's eye and typically ranges from 0 to 100. It serves as a critical metric to gauge the eye's condition during the simulation.
- **Pen Position (pen_position):** The pen's location within the simulation is represented by a two-dimensional array, indicating its x and y coordinates. Accurate tracking of the pen's position is crucial for diagnostic accuracy.
- **Diagnosis (diagnosis):** This string variable will store the final diagnosis reached by the simulation.

Additionally, we establish a loop that acts as the core of the simulation. This loop will persist until specific termination conditions are met.

2. **Simulation Loop:**

Within the simulation loop, several critical steps occur:

- **Display Interface:** The simulation interface is displayed, encompassing a representation of the patient's eye. This interface dynamically updates to reflect changes in the patient's eye health (patient_eye_health) and the current position of the diagnostic pen (pen_position). Realistic graphics and visualizations are essential for an immersive experience.
- **User Input:** User interaction is essential for controlling the diagnostic pen's movement. Input methods used id mouse pointer, enable users to navigate the pen within the simulation.
- **Pen Position Update:** Based on user input, the algorithm updates the pen's position (pen_position). Depending on the simulation's level of sophistication, realistic physics

and graphics are incorporated to accurately mimic the pen's movement and interaction with the virtual eye.

- **Diagnosis Simulation:** The heart of the simulation lies in the diagnosis process: The algorithm determines which region of the eye the pen is currently interacting with, leveraging the pen's position (pen_position). Various diagnostic procedures are simulated. These may involve algorithms to detect abnormalities, analyze the eye's response to stimuli. The realism of these algorithms is crucial for the accuracy of the diagnosis.
- **Health Update:** The patient's eye health (patient_eye_health) is updated to reflect the cumulative effects of diagnostic procedures. In a realistic simulation, this could entail gradual health degradation over time or specific changes based on the diagnostic actions performed.

3. **Simulation Termination:**

To conclude the simulation, we define conditions for its termination:

- **Termination Conditions:** The simulation may end under specific circumstances, as when a conclusive diagnosis is made.
- **Final Diagnosis:** The algorithm determines the final diagnosis by considering the cumulative diagnostic procedures and observations made throughout the simulation. This diagnosis serves as the outcome of the simulation.

4. **End of Simulation:**

- Conclusion: The simulation reaches its conclusion by presenting the user with the final diagnosis. This diagnosis is based on a comprehensive evaluation of the virtual eye's condition and the diagnostic actions performed during the simulation.

4.4 System Interface

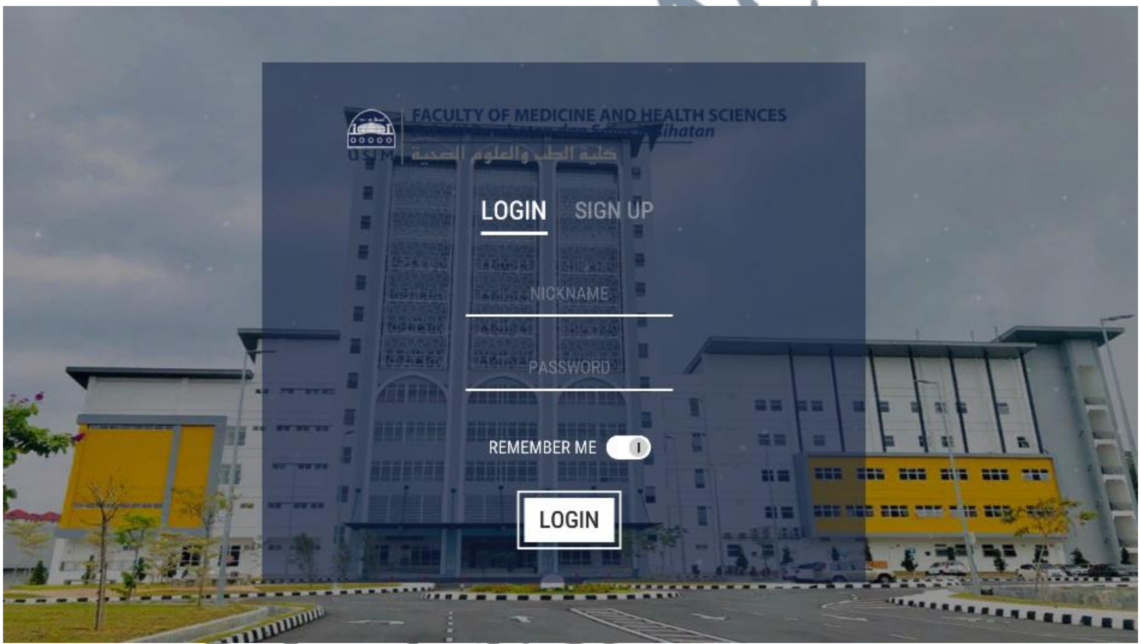

Figure 4. 5 User Login

Figure 4.6 illustrates the user interface for logging into the patient simulator software. To begin using the application, users are required to provide their username and password. Once the necessary credentials have been entered, users can proceed by selecting the login button, which will grant them access to the game interface.

In the event that a user does not possess an existing account, the login interface offers the option to register before being able to engage with the game. By opting to register, users are directed to a separate process where they can supply the essential information needed to establish a new account. This typically entails creating a unique username, setting a secure password, and potentially providing additional details for user identification purposes.

The login user interface within the patient simulator software assumes a pivotal role in ensuring secure and authorized entry to the game. By necessitating the input of valid credentials, the interface acts as a gatekeeper, permitting only registered and authenticated users to proceed. This login process adds an extra layer of protection and accountability, contributing to a more controlled and safeguarded gaming environment.

The login user interface embedded within the patient simulator software holds significant importance in user authentication and access management. Through the provision of username and password, users gain entry to the game after successful verification. Furthermore, the interface accommodates new users by offering a seamless registration option, enabling them to create an account and partake in the immersive patient simulation experience.

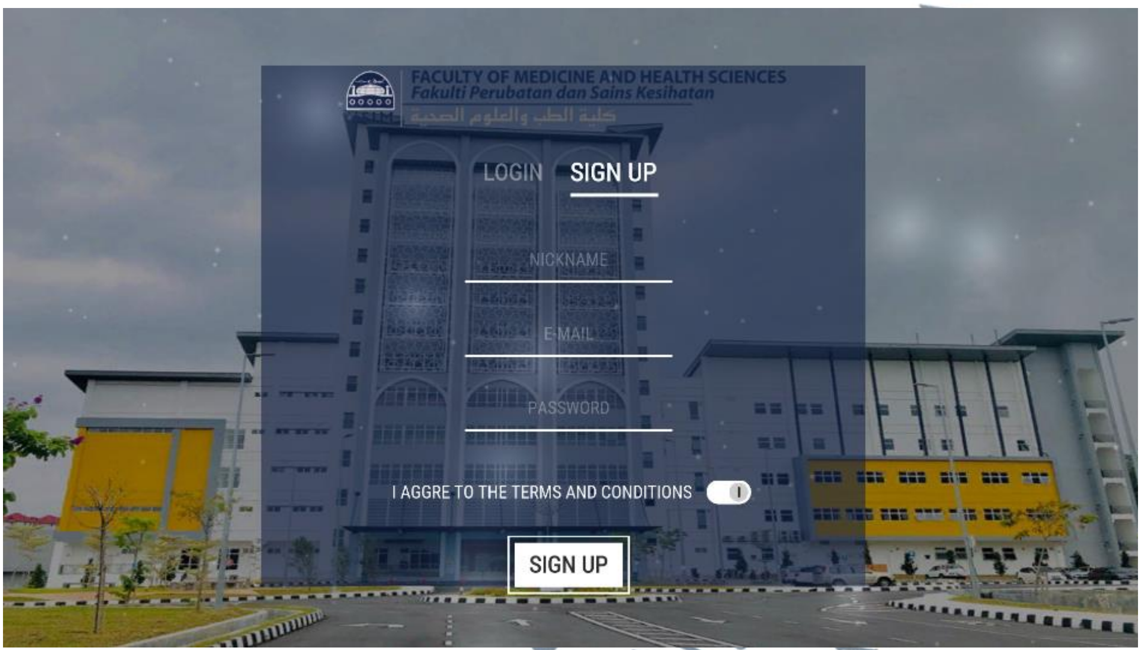

Figure 4. 6 Registration Page

In Figure 4.7, we can observe the registration page of the patient simulator software, which serves as a pivotal step for users to create an account and access the complete range of application features. To successfully register, users are required to diligently provide all the requested information as indicated on the page. This typically includes personal details such as name, email address, and any additional pertinent information necessary for account creation.

Once users have entered all the required information, they are prompted to press the register button, initiating the process of securely storing the provided data within the software's database. By pressing the register button, users grant their consent for the system to retain and utilize the submitted information for account creation and subsequent interactions within the patient simulator software.

The registration page plays a crucial role in ensuring the accuracy and integrity of user data, while also maintaining the security of the software's database. It acts as a pivotal checkpoint to verify that all necessary information is captured and stored correctly. This meticulous registration process significantly contributes to the overall user experience by establishing a reliable foundation for user identification and enabling personalized engagement within the patient simulator software.

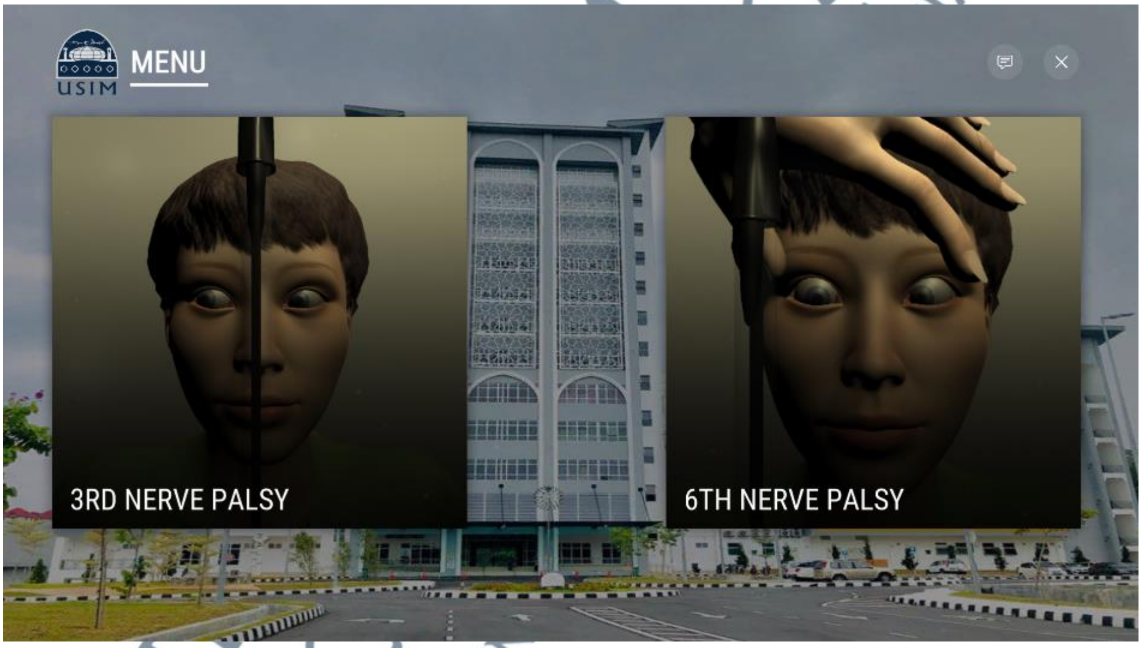

Figure 4. 7 Main Menu UI

In Figure 4.8, we are presented with the patient simulator software's selection scene, which allows users to choose and assess patients with specific disabilities, fostering a realistic and engaging learning experience. Within this scene, users encounter two distinct patient disabilities: 3rd Nerve Palsy and 6th Nerve Palsy. These disabilities serve as authentic scenarios that prompt users to carefully analyse patients' symptoms and medical backgrounds, enabling accurate assessments and informed decisions.

By offering a range of patient disabilities, the patient simulator software aims to provide a comprehensive educational platform for medical students and professionals. Each disability presents unique challenges and opportunities, encouraging users to apply their theoretical knowledge and critical thinking skills within an immersive virtual environment.

The selection scene serves as a gateway to interactive and educational experiences, allowing users to enhance their clinical reasoning, decision-making, and diagnostic capabilities. Through active participation in analysing and diagnosing patients with specific disabilities, users gain valuable hands-on experience and refine their skills in a controlled and safe digital realm.

Figure 4. 8 Virtual Patient Demonstration

Figure 4.9 showcases the dynamic interaction between players and the virtual patient in the patient simulator software. An intriguing feature depicted in the image is the synchronized movement of the patient's eyes with the virtual pen controlled by the user. As the user moves their cursor, the pen mimics the motion, and the software accurately calculates the corresponding eye movement of the virtual patient.

This interactive element serves multiple purposes within the patient simulator software, elevating user engagement by simulating real-world scenarios where eye tracking plays a vital role in medical examinations and assessments. By observing the patient's eye movements in response to their actions, users can develop and refine their observational and diagnostic skills.

Moreover, this feature contributes significantly to the educational value of the patient simulator software, offering users an immersive and hands-on learning experience. Through the seamless interaction between the virtual pen and the patient's eyes, users can practice and enhance their proficiency in conducting eye-related examinations, such as assessing gaze patterns, identifying abnormalities, or evaluating neurological responses.

The integration of the virtual pen and the patient's eye movement exemplifies the advanced capabilities of the patient simulator software, providing a realistic and immersive training environment. By accurately calculating and responding to user inputs, the software creates an authentic simulation that enables users to refine their clinical skills and deepen their understanding of patient interactions.

4.4 Simulation Mechanics

In the field of medical education, simulation has proven to be an invaluable tool for training future healthcare professionals. Simulators allow students to practice and refine their skills in a controlled environment before they interact with real patients. This essay evaluates a specific scenario in a patient simulator involving eye diagnosis, where students are tasked with moving a pen to various positions on the patient's side and lifting the patient's eyelid as required. This evaluation not only assesses the technical proficiency of the students but also underscores the importance of following precise instructions and attention to detail in medical training. Figure 4.10 shows the evaluation or assessment of patient simulation.

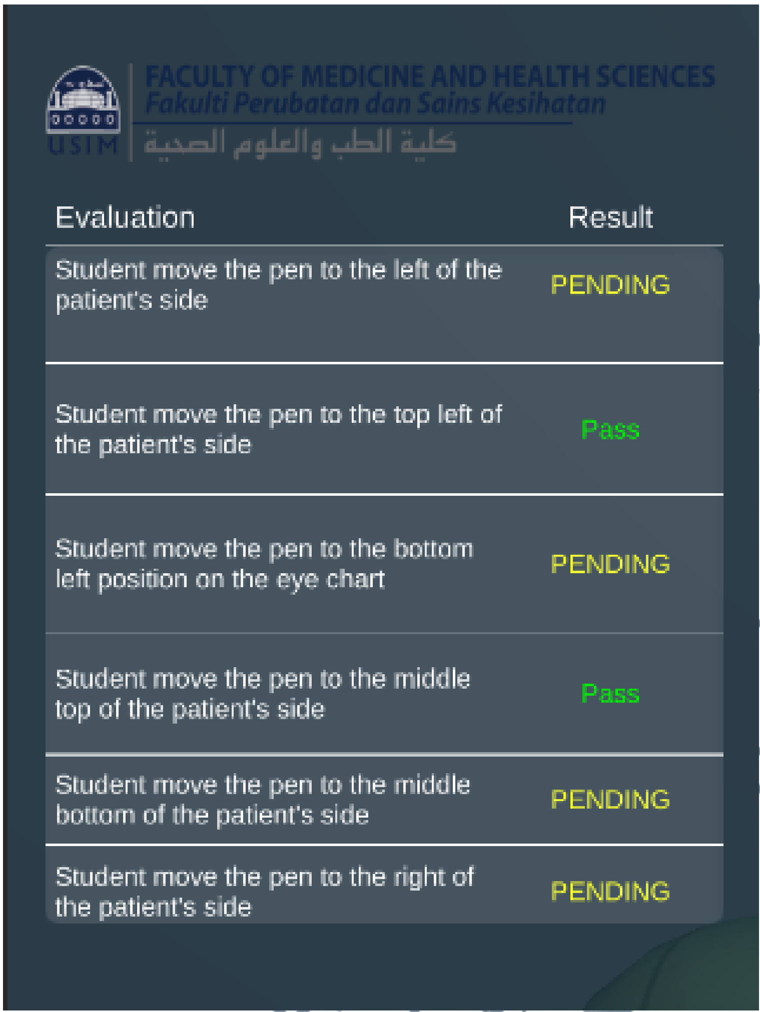

Figure 4. 9 Evaluation

Figure 4. 10 Virtual Patient

The first aspect of the evaluation pertains to the technical proficiency of the students. In steps 1 through 9, students are required to move the pen to specific positions on the patient's side, corresponding to different points on an eye chart. These 9 steps include:

1. Students move the pen to the left of the patient's side.
2. Students move the pen to the top left of the patient's side.
3. Students move the pen to the bottom left position on the eye chart.
4. Students move the pen to the middle top of the patient's side.
5. Students move the pen to the middle bottom of the patient's side.
6. Students move the pen to the right of the patient's side.
7. Students move the pen to the top right of the patient's side.

8. Students move the pen to the bottom right of the patient's side.
9. Students lift up the patient's eyelid to examine the eye.

This demands precision and dexterity, as even a minor deviation from the instructed position could affect the accuracy of the diagnosis. Success in this part of the evaluation is indicative of the student's ability to perform delicate and precise movements, which is crucial in real-world medical examinations.

The evaluation also assesses the student's attention to detail. Each step in the list specifies a precise position to which the pen should be moved. Failure to follow these instructions exactly can result in an inaccurate diagnosis. This underscores the critical importance of attention to detail in medical practice, as even minor errors can have serious consequences for patients. Students who are meticulous in their execution of these steps demonstrate a commitment to patient safety and the accuracy of their diagnoses.

While the first eight steps involve precise movements, the ninth step presents a different challenge. In this step, students are required to lift the patient's eyelid to examine the eye. This step not only tests their technical skills but also their critical thinking and adaptability. They must recognize when and where to apply this action, demonstrating their ability to think on their feet and respond to the specific needs of the patient. This aspect of the evaluation assesses the student's capacity to apply their knowledge in a dynamic clinical setting, where deviations from standard procedures may be necessary.

Figure 4. 11 Evaluation Result UI

In addition to technical proficiency and attention to detail, successful completion of this evaluation also hinges on effective communication and interpersonal skills. Students need to interact with the patient, explaining their actions and ensuring the patient's comfort throughout the process. This interaction reflects their ability to build rapport with patients and convey information in a clear and empathetic manner, which is essential in real clinical practice.

In conclusion, the evaluation of student performance in a patient simulator eye diagnosis scenario assesses various facets of their readiness for clinical practice. It evaluates their technical proficiency, attention to detail, critical thinking, adaptability, and interpersonal skills. Success in this evaluation not only indicates their ability to perform precise movements and follow instructions but also highlights their commitment to patient safety and the accuracy of their diagnoses. Ultimately, it underscores the comprehensive nature of medical training, preparing students to excel in the multifaceted demands of real-world healthcare settings.

4.6 Assessment Actions Dataset

Nerve palsies involving the cranial nerves can lead to a variety of eye movement disorders and visual impairments, often requiring detailed assessments for diagnosis and treatment planning. This essay explores two crucial assessments - one for 3rd nerve palsy and another for 6th nerve palsy - which are critical components of comprehensive eye examinations. We have organized the assessments into tables to provide a clear and systematic understanding of how healthcare professionals evaluate these conditions.

Table 4. 1 6th Nerve Palsy Actions Dataset

Action	Description
Students move the pen to the left of the patient's side	Assess the patient's ability to look to the left (lateral gaze) and check for any limitation or impairment. This may be particularly relevant because the abducens nerve controls the lateral rectus muscle, which is responsible for moving the eye outward.
Students move the pen to the top left of the patient's side	Evaluate the patient's ability to gaze upward and to the left, checking for any deviations or difficulty in these movements.

Students move the pen to the bottom left position on the eye chart	Test the patient's capacity to gaze downward and to the left, which can help identify any limitations in these movements caused by 6th nerve palsy.
Students move the pen to the middle top of the patient's side	Assess the patient's ability to look straight upward to evaluate vertical eye movements.
Students move the pen to the middle bottom of the patient's side	Evaluate the patient's ability to look straight downward to check for any vertical gaze impairments.
Students move the pen to the right of the patient's side	Assess the patient's ability to look to the right (lateral gaze to the right) and identify any limitations or deviations caused by 6th nerve palsy.
Students move the pen to the top right of the patient's side	Evaluate the patient's ability to gaze upward and to the right, checking for any abnormalities in these movements.
Students move the pen to the bottom right of the patient's side	Test the patient's capacity to gaze downward and to the right to identify any limitations in these movements associated with 6th nerve palsy.
Students lift up the patient's eyelid to examine the eye	Examine the patient's eye condition, including checking for any associated ptosis (drooping eyelid), assessing pupil size, and evaluating the overall eye health.

The table Assessment Actions for 6th Nerve Palsy focuses on the assessment process for 6th nerve palsy, also known as abducens nerve palsy. This condition affects the sixth cranial nerve, which primarily controls lateral eye movement. In the case of 6th nerve palsy, students are taught to assess lateral eye movement control and identify limitations or deviations in horizontal and vertical gaze directions. By conducting these assessments and examining the patient's eyelid condition, students gain valuable insights into the location and severity of nerve damage, enabling them to develop informed diagnostic hypotheses.

Table 4. 2 3rd Nerve Palsy Actions Dataset

Action	**Description**
Students move the pen to the left of the patient's side	Assess the patient's ability to look to the left (impairment in lateral eye movement).
Students move the pen to the top left of the patient's side	Evaluate the patient's ability to gaze upward and to the left.
Students move the pen to the bottom left position on the eye chart	Test the patient's ability to gaze downward and to the left.
Students move the pen to the middle top of the patient's side	Assess the patient's capacity to look straight upward.
Students move the pen to the middle bottom of the patient's side	Evaluate the patient's ability to look straight downward.

Students move the pen to the right of the patient's side	Assess the patient's ability to look to the right (impairment in lateral eye movement to the right).
Students move the pen to the top right of the patient's side	Evaluate the patient's ability to gaze upward and to the right.
Students move the pen to the bottom right of the patient's side	Test the patient's ability to gaze downward and to the right.
Students lift up the patient's eyelid to examine the eye	Examine for ptosis (drooping eyelid) and assess the overall eye condition, including pupil size and response to light.

The table Assessment Actions for 3rd Nerve Palsy outlines a series of actions a healthcare provider might perform when examining a patient suspected of having 3rd nerve palsy, also known as oculomotor nerve palsy. This condition affects the third cranial nerve, which plays a pivotal role in controlling eye movement and pupil dilation. The assessment actions encompass a range of eye movements and gaze directions. Students learn to evaluate the patient's ability to control eye movement, identify abnormalities in gaze directions, and examine associated factors such as eyelid conditions and pupil responses. These actions collectively equip students with the skills necessary to assess the extent of nerve damage and contribute to accurate diagnoses.

In conclusion, these tables provide a structured learning for students to follow when diagnosing patients with 3rd and 6th nerve palsy. By mastering these assessment actions, students not only hone their clinical skills but also contribute to the early detection and accurate diagnosis of cranial nerve palsies. Ultimately, the knowledge and proficiency gained through these assessments empower students to become competent healthcare professionals capable of delivering targeted care and improving patients' lives.

4.7 Game Security Implementation

The advent of technology has revolutionized various sectors of the healthcare industry, and one such area is the development of eye diagnosis simulations. These simulations play a pivotal role in training healthcare professionals, students, and researchers in understanding eye conditions and their treatment. However, with the increasing integration of technology into the healthcare domain, it becomes imperative to prioritize user database security. This essay explores the significance of a secure database login system and registration process, with a focus on hashing techniques, within the context of an Eye Diagnosis Simulation platform.

1. **Importance of User Database Security:**

Confidentiality of Patient Data

In an Eye Diagnosis Simulation platform, user data primarily includes patient information, medical history, and diagnostic records. Ensuring the confidentiality of this sensitive data is paramount. Unauthorized access to patient data can lead to privacy breaches, legal implications, and a loss of trust in the platform. A secure database system is essential to safeguard patient information.

Protection against Unauthorized Access

User database security is not only about protecting patient data but also about preventing unauthorized access to the simulation platform. Unauthorized users, such as hackers, can potentially disrupt the system, compromise its integrity, and even exploit the resources. To prevent this, a robust login and registration system is required.

2. **Password Security:** Hashing

Hashing is a fundamental technique used in securing user passwords within a database. It involves converting a plain text password into a fixed-length string of characters, known as a hash. Hashing is a one-way process, meaning it cannot be reversed to reveal the original password, making it a suitable choice for protecting user credentials.

3. **Secure Registration Process**: Password Complexity Requirements

During the registration process, users should be encouraged to create strong passwords. This includes a mix of uppercase and lowercase letters, numbers, and special characters. Implementing password complexity requirements helps mitigate the risk of weak and easily guessable passwords.