Augmented and Virtual Reality Applications

Final Report

Group E

Hang Yu	hay0583@thi.de
Sara Taleb Nezhad	sat0895@thi.de
Waleed Binsaad	wab3680@thi.de

Meta Data

About Us

- **Waleed:**

  I am a skilled programmer with extensive experience in software development. My practical mindset and ability to bring ideas to life were invaluable to this project.

  In this project, I primarily took on the role of developer. I implemented the Hints in AR, developed the 3D puzzle game for Mini-Game 2, and created the simulated printing game for Mini-Game 3. Additionally, I was responsible for integrating all task modules into a cohesive system.

- **Hang:**

  I am a structured and detail-oriented professional with a background in product design, project management, and technology. My experience as a product owner at several internet companies in China has provided me with extensive expertise in product management and feature design.

  My main role in this project was project management. I planned the game concept and timeline, documented all ideas and discussions, and ensured tasks were completed on schedule. I also developed the 2D puzzle game for Mini-Game 1. During the testing phase, I conducted quantitative analysis, including HARUS results, to evaluate user feedback and make iteration plans.

- **Sara:**

  I am a UX-focused designer with a Bachelor's degree in Bioelectrical Engineering and experience as an electronics researcher and in biomedical engineering. I have designed websites and applications for projects in Iran and Germany, including a mobile app for the Car2X Lab at THI University and the shift handover process for Klingele Factory as a UI and UX designer.

  I worked as a UX and UI designer on this project, responsible for translating the game concept and user requirements into low-fidelity and high-fidelity designs. I iterated the UI design based on feedback from multiple testing sessions to enhance the user experience. Additionally, I managed all user testing phases, conducted qualitative analysis, and collected insights from user interviews and feedback recordings.

GitHub Link

https://github.com/wbinsaad/DiscoverTHI

How To Start This Project

When you launch the app, you are automatically taken to the maps page, where you can see all the tasks available. Each task is clearly displayed on the map, making it easy to navigate through the different challenges.

To get started, simply click the 'Hint 1' button on the map to start the first task: find the AR backpack. As you progress, the app will provide step-by-step guidance through pop-up notifications to help you stay on track and transition smoothly to the next task.

For Mini-Game 2, you will need to print out the four QR codes (see Appendix B) and place them on a table. Scan each of the QR codes and you will see four pieces of the 3D puzzle appear on your screen. Next, move the QR codes around the table to complete Mini-Game 2.

Mini-Game 3 requires you to scan another QR code (see Appendix C), which will bring up a virtual printer and student ID on your screen. Once these are displayed, you can start Mini-Game 3 and progress through the challenges.

Content

1. INTRODUCTION

1.1 Project Overview

As campus life becomes increasingly digital, new students often face challenges in adapting to campus resources and facilities. For many, mastering the use of basic campus tools such as printers and navigation systems is an essential part of their transition into university life. However, traditional orientation methods often fail to provide an engaging, hands-on learning experience.

"Discover THI" is an AR game designed to help new students at the THI quickly familiarize themselves with essential campus resources. The game offers an interactive, immersive environment where students can learn how to use campus facilities such as printers, navigate the campus using AR, and solve puzzles related to campus life.

By gamifying this process, **"Discover THI"** aims to make learning about campus resources both fun and educational, reducing the anxiety many new students feel when navigating unfamiliar environments. Through this project, we aim to enhance the overall student experience, helping them integrate into campus life with ease, while also promoting the use of digital tools in educational settings.

1.2 Project Objectives

The main goal of the project is to design and develop an interactive AR-based game, that helps new students quickly adapt to campus life and effectively utilize campus resources. The specific objectives of the project are as follows:

- **Enhance User Experience:** Provide an engaging, immersive game experience that allows students to learn how to use campus resources in a fun, interactive way. The goal is to reduce the monotony of traditional learning methods and alleviate the stress new students often experience.
- **Achieve Technical Goals:** Implement an AR-based campus navigation system that enables users to interact with campus resources using their phone's camera and receive real-time feedback. Optimize the game interface to ensure smooth and intuitive operation, especially in terms of AR object detection and interaction.
- **Meet User Needs:** Tailor the game for new students at THI to help them quickly become familiar with essential campus facilities such as printers. Through game elements like hints

and mini-games, guide users step by step in mastering the necessary skills, while enhancing their sense of achievement with features like a virtual backpack or a student card.

- **Address Key Problems:** Challenges in adapting to campus resources: New students often feel overwhelmed by the unfamiliarity of campus facilities and may struggle with initial usage. Barriers to information access: Traditional orientation methods, such as printed guides or static maps, are not engaging and may not effectively communicate how to use campus resources.

2. Project Concept

2.1 User Stories

As a new student at THI: I want to quickly learn how to use campus resources, such as the printer, through an interactive game, so that I can complete basic tasks on my first day without difficulties. Through the game's hints and mini-games, I can gradually learn how to perform essential tasks on campus, particularly how to print my timetable. This step-by-step guidance will help me feel more confident and prepared.

As a student interested in AR: I want to gradually unlock different stages of the game using virtual items, so that I can satisfy my curiosity and obtain important campus information in an engaging way. The game design uses a series of puzzles and tasks that progressively guide me through essential campus operations. This provides an immersive and fun experience, allowing me to learn while exploring the campus interactively.

2.2 Personas

Persona 1 - Hayley the New Student: Hayley is a 18-year-old new student at THI, feeling excited but also a bit overwhelmed by the new campus environment. She is tech-savvy and loves interactive experiences, but she feels uncertain about the day-to-day tasks at university.

Needs	● To quickly get familiar with the campus and its services. ● To learn how to perform important tasks, like printing her timetable, in a fun and engaging way.

Motivations	● To feel more confident navigating her new environment. ● To save time and avoid feeling lost on her first day.
Behavior Traits	● Enjoys learning through gamified experiences. ● Prefers step-by-step guidance but likes to be challenged.

Persona 2 - Daniel the Non-Techie: Daniel is not very comfortable with technology and often finds new systems a bit confusing. He is a 19-year-old international student at THI and worries about using the campus's digital systems, particularly for things like printing.

Needs	● Clear instructions on how to use campus equipment and resources. ● To be eased into using digital tools without feeling overwhelmed.
Motivations	● To quickly understand the process of using campus printers without feeling intimidated. ● To gain confidence in using the technology provided by the university.
Behavior Traits	● Prefers simple, hands-on activities that offer clear, visual instructions. ● Likes repetitive, practice-based learning to build familiarity.

Persona 3 - Mark the Gamified Learner: Mark is a 22-year-old master student who enjoys playing video games and is used to interactive learning methods. He appreciates games that incorporate real-world tasks and finds them exciting and effective.

Needs	● To learn new tasks (like printing) in an immersive and fun way. ● To feel challenged and rewarded for completing the game.
Motivations	● To master practical skills while enjoying the gameplay experience. ● To feel a sense of accomplishment through the game’s progression.
Behavior Traits	● Engages well with puzzles and challenges. ● Prefers to learn by doing rather than reading instructions.

2.3 Design Considerations

For UI/UX design, the primary focus was on creating an intuitive, engaging, and seamless user experience while ensuring the interface supports users' needs effectively. Here are the key considerations and features integrated into the design:

Level Map Integration: A level map was added as a menu that users can access both at the beginning of the game and during gameplay. Designed based on user feedback, the map starts from the bottom, making navigation more intuitive and aligning with players' mental models of progression.

Pre-Hint Information: Before each hint, a pop-up was introduced to guide players by explaining what they should find and how to proceed. This feature minimizes confusion and ensures that players remain engaged and informed throughout the gameplay.

Information Accessibility: A dedicated "Information" button was added to the game page, enabling users to quickly access relevant guidance if they forget what to do. This ensures players can easily retrieve important instructions without disrupting their gaming experience.

Feedback and Failure Handling: An "Oops" pop-up was designed to appear when players fail to complete a game, providing clear feedback and requiring them to retry the most recent game. This feature was included to clarify progression rules and encourage users to improve their performance.

Camera Hint Guidance: For games that utilize camera functionality (games 2, 3, and certain hints), a short information message is displayed as the camera starts. This helps users understand how to interact with the feature, reducing uncertainty and improving the overall experience.

Color Palette and Visual Design: The yellow and purple color palette was chosen for its strong visual contrast and vibrant appeal. These colors create a distinctive look while maintaining usability and enhancing visibility across different elements of the interface.

Touch-Friendly Controls: To accommodate touch input, we implemented large, well-spaced controls with clear feedback mechanisms. These elements ensure ease of use and reduce errors, especially for mobile users.

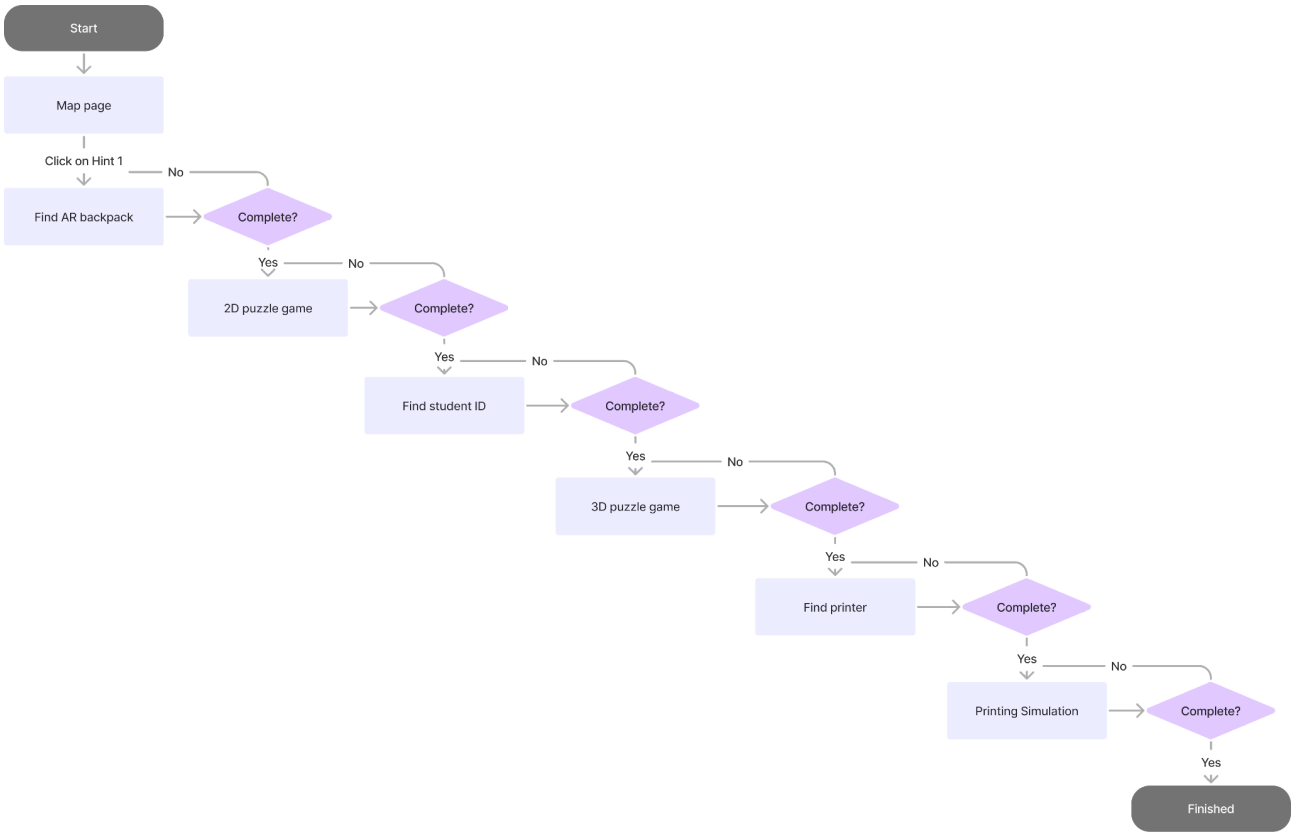
Iterative Design Process: Regular user feedback sessions played a crucial role in shaping the final design. Iterative updates were made to refine the user experience and address player concerns, ensuring the interface aligns with user expectations.

By combining functional elements with a user-centered approach, the design strikes a balance between usability, aesthetics, and performance. Every feature and interaction was carefully crafted to provide an engaging and accessible experience for a diverse audience.

For detailed final design screens and Figma link, please refer to Appendix A: Hi-Fi Design.

3. Planned User Activities

The user flow of "Discover THI" is designed to guide players through a series of interconnected tasks in a linear progression. Starting from the Map Page, players navigate through three Hints and their corresponding Mini-Games, with each step unlocking the next task upon successful completion. Here's a diagram of the entire user flow.



Below is a detailed breakdown of the user flow:

The Map Page (The Starting Point): The Map Page serves as the main navigation hub and progress tracker for the game. It displays the sequence of tasks in a clear and intuitive format: Hint 1 → Mini-Game 1 → Hint 2 → Mini-Game 2 → Hint 3 → Mini-Game 3. Players can revisit the Map Page at any time to check their progress and identify which tasks remain.

Hint 1 (Find the backpack): The journey begins with Hint 1, where players are tasked with locating a virtual Backpack in an AR environment:

- A 3D backpack object randomly appears on a flat surface.
- Players use their phone's camera to scan the environment, locate the backpack, and tap on it to collect puzzle pieces.

After completing this task, a pop-up window prompts players to proceed to Mini-Game 1.

Minigame 1 (2D Puzzle): In the first mini-game, players solve a 2D puzzle:

- Drag-and-drop puzzle pieces into the correct positions within a frame.
- Upon completing the puzzle, an image of a Student ID Card is revealed.

This student card becomes the target for the next task. Players are prompted to move on to Hint 2.

Hint 2 (Find the Student ID): The second hint closely mirrors the process of Hint 1:

- A virtual Student ID Card appears somewhere in the AR environment.
- Players use their phone's camera to scan, locate, and tap on the card.

Once the card is found, a pop-up window directs players to proceed to Mini-Game 2.

Minigame 2 (3D Puzzle): In this mini-game, players solve a 3D puzzle using a QR code mechanism:

- Four QR codes are physically placed on a table in the real world.
- Players scan the QR codes with their phone's camera to reveal small 3D puzzle pieces (4 cubes).
- By physically rearranging the QR codes on the table, players assemble the puzzle into a cohesive 3D object.

Upon completing the puzzle, an image of a Printer appears, signaling the target for the next task. Players are then prompted to move on to Hint 3.

Hint 3 (Find the printer): The third hint follows the same structure as the previous hints:

- A virtual Printer appears in the AR environment.
- Players use their phone's camera to locate and tap on the printer.

After finding the printer, a pop-up window notifies players that Mini-Game 3 is now available.

Minigame 3 (Simulate a Print Task): In the final mini-game, players simulate a print action:

- Players scan a QR code to locate the virtual Printer.
- A Student ID Card appears on the screen and moves horizontally.
- Players must tap the screen when the card aligns with the printer, simulating a print task.

Once the task is successfully completed, the game concludes.

4. Methods for Content Creation

4.1 Development Methodology

The development of "Discover THI" followed an Agile approach, ensuring flexibility and adaptability throughout the process. Notion was used to assign tasks, track progress, and maintain transparency within the team. The workflow was divided into five key stages:

- **Concept and Requirement Analysis:** The team defined the project goals and identified the target audience, ensuring alignment with user needs.
- **Prototyping:** Initial prototypes were created to test the core game concept, validate the user flow, and refine ideas based on early feedback.
- **Implementation:** The core functionalities of the game were developed, with AR features seamlessly integrated to create an immersive user experience.
- **Testing and Iteration:** Feedback from user testing sessions was gathered to identify pain points and optimize the game design and mechanics.
- **Deployment:** Final adjustments and performance optimizations were made to prepare the game for release.

GitHub was utilized for version control and code management, enabling efficient collaboration and maintaining code integrity. Team communication was facilitated through weekly meetings and online tools such as Figma and Notion, ensuring smooth coordination and effective problem-solving.

4.2 Class & Entities

Enum: Levels

Description: Defines the different levels or stages within the game.

Values:

- hint1: Represents the first hint stage.
- hint2: Represents the second hint stage.
- hint3: Represents the third hint stage.
- game1: Represents the first game stage.
- game2: Represents the second game stage.
- game3: Represents the third game stage.

Class: UserProfile

Description: Represents a user profile within the game, storing the user's name and current level.

Member Variables:

- Name (string): The user's name.
- Level (Levels): The user's current level in the game.

Constructors:

- UserProfile()
    - Signature: public UserProfile()
    - Description: Default constructor. Initializes a new UserProfile object with an empty string for Name and Levels hint1 since it’s the starting level.
- UserProfile(string Name, Levels Level)
    - Signature: public UserProfile(string Name, Levels Level)
    - Description: Overloaded constructor allowing you to specify the user's name and initial level during object creation.

Class: GameManagerController

Description: A singleton class that manages the game's overall state, user profile, and persistent data storage. It also handles background music and sound effects.

Member Variables:

- Instance (static GameManagerController): A reference to the single instance of this class, following the singleton pattern.
- UserProfile (static UserProfile): A reference to the user profile object containing the user's name and current level.
- path (string): The file path to the persistent data storage file (game.json).

- BackGroundClip (AudioClip): The audio clip for the background music.

Methods:

- Awake()
    - Signature: void Awake()
    - Description: Sets up the singleton instance. If no instance exists, it creates one and assigns it to Instance. The DontDestroyOnLoad method ensures this game object persists across scene loads. If another instance already exists, the current game object is destroyed to enforce the singleton pattern.
- Start()
    - Signature: void Start()
    - Description: Starts playing the background music in a loop using the SoundController class.
        - Sets the path to the persistent data storage file (game.json) by combining the application's persistent data path with the filename.
        - Requests permission to write to external storage for saving the user profile.
        - Calls the GetCurrentUser method to load the user profile data.
- UpdateUserProfile(Levels updatedLevel)
    - Signature: public void UpdateUserProfile(Levels updatedLevel)
    - Description: Updates the user's current level in the persistent data file.
        - Checks if the data file exists.
        - If the file exists, creates a new UserProfile object with the user's name (assuming it's always "User") and the provided updatedLevel.
        - Converts the user profile object to a JSON string using JsonUtility.ToJson.
        - Writes the JSON string to the data file using File.WriteAllText.
- GetCurrentUser()
    - Signature: public UserProfile GetCurrentUser()
    - Description: Loads the user profile data from the persistent storage file.
        - Checks if the data file exists.
        - If the file exists, reads the entire file content as a string using File.ReadAllText.
        - Converts the JSON string back to a UserProfile object using JsonUtility.FromJson.
        - Returns the loaded user profile.

- If the file doesn't exist, creates a new default user profile with the name "User" and the starting level (Levels.hint1), saves it to the data file, and returns the newly created profile.

Class: SceneController

Description: Provides a simple method for switching between scenes in the game.

Methods:

- SwitchScenes(string sceneName)
  - Signature: public void SwitchScenes(string sceneName)
  - Description: Loads the scene specified by the sceneName parameter using SceneManager.LoadScene.

Class: HomeController

Description: Manages the UI elements on the home screen, specifically the visual state (active, finished, or locked) of the hint and game buttons based on the user's current game progress.

Member Variables:

- ActiveButton (Sprite): The sprite used to represent an active (available) button.
- FinishedButton (Sprite): The sprite used to represent a finished (completed) button.
- LockButton (Sprite): The sprite used to represent a locked (unavailable) button.
- Hint1, Hint2, Hint3 (GameObject): Game objects representing the hint buttons for each level.
- Game1, Game2, Game3 (GameObject): Game objects representing the game buttons for each level.
- hints (GameObject[]): An array holding the hint game objects for easier iteration.
- games (GameObject[]): An array holding the game game objects for easier iteration.

Methods:

- Start()
  - Signature: void Start()
  - Description: Initializes the hints and games arrays. Retrieves the current user profile from the GameManagerController and calls UpdateUIBasedOnLevel to set the initial button states.

- UpdateUIBasedOnLevel(Levels currentLevel)
    - Signature: private void UpdateUIBasedOnLevel(Levels currentLevel)
    - Description: Updates the button states based on the provided currentLevel.
        - Iterates through the hint and game buttons.
        - Uses the currentLevel value and the loop index i to determine the state of each button:
            - Finished: If (int)currentLevel > i * 2 + 1, both the hint and game buttons for the current level are set to the FinishedButton sprite.
            - Active Hint: If (int)currentLevel == i * 2 + 1, the hint button is set to the ActiveButton sprite, and the corresponding game button is set to LockButton.
            - Active Game: If (int)currentLevel == i * 2 + 2, both the hint and game buttons for the current level are set to the ActiveButton sprite.
            - Locked: In all other cases, both the hint and game buttons for the current level are set to the LockButton sprite.
- SetButtonState(GameObject buttonObject, Sprite stateSprite)
    - Signature: private void SetButtonState(GameObject buttonObject, Sprite stateSprite)
    - Description: Sets the sprite of the provided buttonObject's Image component to the provided stateSprite. Does nothing if buttonObject is null.

Class: SoundController

Description: A singleton class that handles playing sound effects and background music.

Member Variables:

- Instance (static SoundController): A reference to the single instance of this class, following the singleton pattern.
- audioSourceOnce (AudioSource): An AudioSource component used for playing one-shot sound effects.
- audioSourceLoop (AudioSource): An AudioSource component used for playing looping sound effects or background music.

Methods:

- Awake()

    - Signature: void Awake()
    - Description: Sets up the singleton instance.
- PlaySound(AudioClip clip)
    - Signature: public void PlaySound(AudioClip clip)
    - Description: Plays a provided sound effect (clip) once.
- PlaySoundLoop(AudioClip clip)
    - Signature: public void PlaySoundLoop(AudioClip clip)
    - Description: Starts playing a provided sound effect (clip) in a loop.
- StopLoopSound()
    - Signature: public void StopLoopSound()
    - Description: Stops any currently playing loop sound effect or background music.

Class: HintController

Description: Controls the hint object placement and interaction, handling scene transitions upon successful interaction.

Member Variables:

- HintObject (GameObject): The GameObject prefab to be instantiated as the hint.
- NextSceneName (string): The name of the scene to load upon successful hint interaction.
- NextLevel (Levels): The level to update the user profile to upon successful hint interaction.

Methods:

- Start()
    - Signature: void Start()
    - Description: Instantiates the HintObject at a random offset from the camera's position.
- Update()
    - Signature: void Update()
    - Description: Detects touch input and performs a raycast. If the raycast hits an object with the tag "HintObject", triggers vibration, updates the user profile to NextLevel, and loads the scene specified by NextSceneName.

Class: LookAt

Description: Makes the attached GameObject constantly face the main camera for hints.

Methods:

- Update()
    - Signature: void Update()
    - Description: Orients the GameObject to face the main camera's position.

Class: Game1Controller

Description: Manage the game logic for level 1, including tracking placed pieces and handling game completion.

Member Variables:

- Instance (static Game1Controller): A reference to the single instance of this class.
- totalPieces (int): The total number of game pieces.
- placedPieces (int): The number of placed game pieces.

Methods:

- Awake()
    - Signature: void Awake()
    - Description: Sets up the singleton instance.
- CheckGameComplete()
    - Signature: public void CheckGameComplete()
    - Description: Increments placed pieces and checks for game completion.
- ReloadScene()
    - Signature: public void ReloadScene()
    - Description: Reloads the current scene.
- ChangeScnese(string sceneName)
    - Signature: public void ChangeScnese(string sceneName)
    - Description: Loads a new scene.
- OnSuccessButtonClick()
    - Signature: public void OnSuccessButtonClick()
    - Description: Handles success button clicks, triggers vibration, and loads the next scene.

Class: DragImage

Description: Handles dragging behavior for UI images and checks for correct placement.

Member Variables:

- rectTransform (RectTransform): The RectTransform of the draggable image.
- canvasGroup (CanvasGroup): The CanvasGroup of the draggable image.
- originalPosition (Vector2): The original position of the image.
- correctPosition (Vector2): The correct position for the image.
- correctObject (GameObject): The corresponding correct object.
- threshold (float): The Permissible distance to correct position.

Methods:

- Awake()
    - Signature: void Awake()
    - Description: Initializes references and the original position.
- OnBeginDrag(PointerEventData eventData)
    - Signature: public void OnBeginDrag(PointerEventData eventData)
    - Description: Handles the beginning of a drag event.
- OnDrag(PointerEventData eventData)
    - Signature: public void OnDrag(PointerEventData eventData)
    - Description: Handles the drag event.
- OnEndDrag(PointerEventData eventData)
    - Signature: public void OnEndDrag(PointerEventData eventData)
    - Description: Handles the end of a drag event and checks for correct placement.
- CanvasScale()
    - Signature: private float CanvasScale()
    - Description: Gets the canvas scale factor.

Class: PuzzlePiece

Description: Handles touch input and movement for puzzle pieces.

Member Variables:

- correctPosition (Vector2): The correct position for the puzzle piece.
- targetGameObject (GameObject): The target GameObject for correct placement.
- isDragging (bool): Flag indicating if the piece is being dragged.
- offset (Vector2): Offset between touch position and piece position.
- originalPosition (Vector3): The original position of the piece.

- isPlaced (bool): Flag indicating if the piece is placed correctly.
- fingerId (int): Touch ID for Multi-Touch

Methods:

- Start()
    - Signature: void Start()
    - Description: Initializes the correct position and original position.
- Update()
    - Signature: void Update()
    - Description: Handles touch input and piece movement.
- CheckPosition()
    - Signature: void CheckPosition()
    - Description: Checks if the piece is placed correctly.

Class: UIManager

Description: Manages UI elements, including the fail and success panels.

Member Variables:

- Instance (static UIManager): A reference to the single instance of this class.
- failPanel (GameObject): The fail panel game object.
- successPanel (GameObject): The success panel game object.
- successButton (GameObject): The success button game object.

Methods:

- Awake()
    - Signature: void Awake()
    - Description: Sets up the singleton instance.
- ShowFailPanel()
    - Signature: public void ShowFailPanel()
    - Description: Shows the fail panel.
- ShowSuccessPanel()
    - Signature: public void ShowSuccessPanel()
    - Description: Shows the success panel.
- ShowSuccessButton()

- Signature: public void ShowSuccessButton()
- Description: Shows the success button.

Class: Game2Controller

Description: Manages the gameplay logic and AR interactions for the second game scene.

Member Variables:

- ArPrefabs (GameObject[]): An array of AR prefabs representing the game pieces.
- trackedImages (ARTrackedImageManager): A reference to the ARTrackedImageManager component.
- ARObjects (List<GameObject>): A list to store instantiated AR objects.

Methods:

- Start()
  - Signature: void Start()
  - Description: Sets the screen sleep timeout to never sleep.
- Update()
  - Signature: void Update()
  - Description: Checks for the win condition by comparing the positions of the AR objects. If the win condition is met, the scene is changed to "Hint3Scene".
- Awake()
  - Signature: void Awake()
  - Description: Gets a reference to the ARTrackedImageManager component.
- OnEnable()
  - Signature: void OnEnable()
  - Description: Subscribes to the trackedImagesChanged event of the ARTrackedImageManager component.
- OnDisable()
  - Signature: void OnDisable()
  - Description: Unsubscribes from the trackedImagesChanged event of the ARTrackedImageManager component.
- OnDestroy()
  - Signature: void OnDestroy()
  - Description: Destroys all instantiated AR objects and clears the ARObjects list.

- OnTrackedImagesChanged(ARTrackedImagesChangedEventArgs eventArgs)
    - Signature: private void OnTrackedImagesChanged(ARTrackedImagesChangedEventArgs eventArgs)
    - Description: Handles the trackedImagesChanged event by creating new AR objects if a tracked image is added, updating the tracking positions of existing AR objects.
- CheckWinCondition()
    - Signature: private bool CheckWinCondition()
    - Description: Checks if the win condition is met by comparing the positions of the AR objects. The win condition is met if all four AR objects are placed within a certain threshold of their expected positions relative to each other.

Class: Game3Controller

Description: Manages the gameplay logic and AR interactions for the third game scene. This includes instantiating and tracking AR prefabs, animating the movement of a student card around a printer, and detecting successful card placement based on user input and position thresholds.

Member Variables:

- PrinterPrefab (GameObject): The prefab for the printer AR object.
- StudentCardPrefab (GameObject): The prefab for the student card AR object.
- instantiatedPrinter (GameObject): A reference to the instantiated printer object.
- instantiatedStudentCard (GameObject): A reference to the instantiated student card object.
- trackedImageManager (ARTrackedImageManager): A reference to the ARTrackedImageManager component.
- lastPosition (Vector3): Stores the last known position of the tracked image for movement threshold checks.
- positionThreshold (float): The minimum distance the tracked image needs to move before updating the student card position.
- speed (float): The speed of the student card's side-to-side movement animation.
- range (float): The range of the student card's side-to-side movement animation.
- EndPopup (GameObject): The game object representing the end popup UI element.

Methods:

- Start()
    - Signature: void Start()

    - Description: Sets the screen sleep timeout to never sleep to prevent interruptions during gameplay.
- Awake()
    - Signature: void Awake()
    - Description: Gets a reference to the ARTrackedImageManager component.
- OnEnable()
    - Signature: void OnEnable()
    - Description: Subscribes to the trackedImagesChanged event of the ARTrackedImageManager component to handle new and updated tracked images.
- OnDisable()
    - Signature: void OnDisable()
    - Description: Unsubscribes from the trackedImagesChanged event of the ARTrackedImageManager component to avoid memory leaks.
- Update()
    - Signature: void Update()
    - Description: Manages the student card's movement animation and checks for user input to trigger win condition logic.
        - Calculates a side-to-side movement offset for the student card based on speed and range.
        - Transforms the local offset to world space relative to the printer's position.
        - Updates the student card's position and rotation.
        - Checks for user mouse click.
        - If clicked, calls the IsCardOnSameX method to determine if the card is within the X-axis threshold of the printer.
        - If successful, logs a message and activates the EndPopup game object.
- OnTrackedImagesChanged(ARTrackedImagesChangedEventArgs args)
    - Signature: void OnTrackedImagesChanged(ARTrackedImagesChangedEventArgs args)
    - Description: Handles events related to tracked images.
        - If the printer hasn't been instantiated yet, creates a new printer object at the tracked image's position and calls CreateStudentCard to create the student card.
        - Checks the image's tracking state and activates/deactivates the printer and student card objects accordingly.
        - If the image is tracked and the student card exists, calls UpdateStudentCard to update its position if the movement exceeds the threshold.

- CreateStudentCard()
    - Signature: void CreateStudentCard()
    - Description: Creates and positions the student card object relative to the printer.
- UpdateStudentCard(Vector3 position)
    - Signature: void UpdateStudentCard(Vector3 position)
    - Description: Updates the student card's position

5. User Testing

5.1 Testing Methods

5.1.1 Quantitative Testing

- **Time to Completion**: We measured the time it took for participants to complete each task as well as the entire game. Our initial assumption was that each task would take around 30 seconds, and the total game completion time would be quick. This provided a benchmark to assess the game's difficulty and user interaction speed.
- **Breaks-in-Presence**: We tracked when participants paused or stopped interacting with the game. This helped us understand which parts of the game were confusing or frustrating, and where users might be losing focus.

5.1.2 Qualitative Testing

- **HARUS Survey**: After completing the game, participants were asked to fill out the HARUS survey via Google Forms. This survey consisted of 16 statements, divided into two categories: manipulability and comprehensibility. Participants rated their level of agreement with each statement to provide insights into how easy or difficult they found the game to navigate and understand.
- **User Feedback via Interviews**: In addition to the survey, we conducted brief interviews with participants to gather qualitative feedback. During these interviews, we asked participants about their overall experience, any confusion they encountered, and their suggestions for improvements.

5.2 Technical Setup

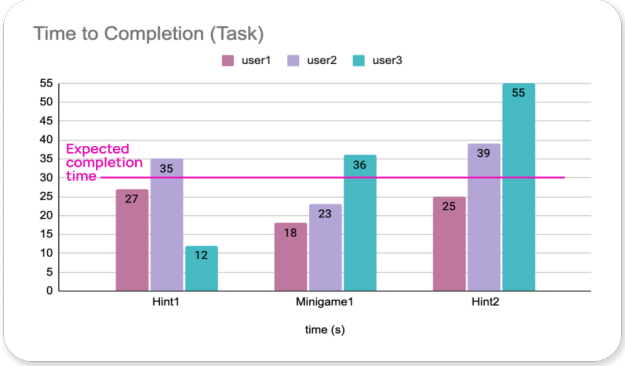
- **Controlled Environment:** All testing was conducted in a controlled classroom environment, ensuring that external factors did not influence the results.
- **Devices:** The game was tested on Android phones, which were selected based on compatibility and availability within the testing group.

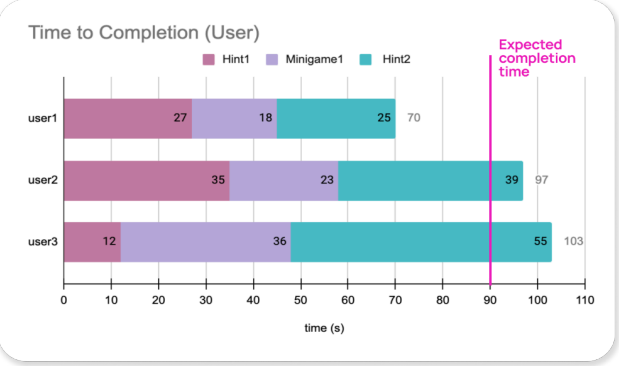
5.3 Testing Results

5.3.1 First Round of User Testing

- **Test Scope**: The testing focused on Hint 1, Mini-Game 1, and Hint 2.
- **Participants**: Three THI students, all of whom are part of our target user group, participated in this round of testing.

Time to Completion:





We initially estimated that each task (Hint 1, Mini-Game 1, Hint 2) would take around 30 seconds to complete. However, we observed that only one user finished Hint 1, Mini-Game 1, and Hint 2 within 90 seconds. The other two users took significantly longer, especially when trying to find AR objects in Hint 1 and Hint 2. This suggests that the tasks may be more challenging than anticipated, and that the AR interaction was not as intuitive as we expected.

Breaks-in-Presence:

We noted several points where participants paused their interactions:

- **Hint 1** – Finding the AR Backpack: One user struggled because they didn't know what the target object (the backpack) looked like. Another user didn't realize that they had to pick up

the phone and rotate the camera, so they left the phone on the table and didn't interact with it.

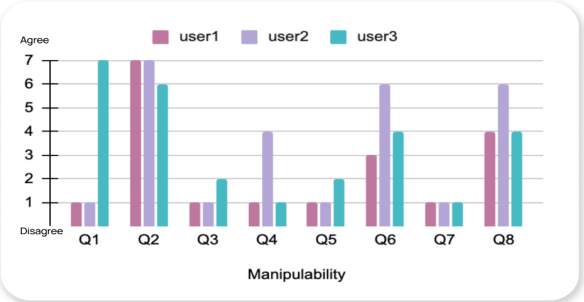
- **Mingame 1** – 2D Puzzle: Two users wanted to return to the previous page for more information but could not find the back button, causing frustration.
- **Hint 2** – Finding the Student Card: During this hint, the AR system took too long to detect the plane surface, leading to frustration. Additionally, a bug prevented the student card from being clickable, which caused the participants to stop at this point.

HARUS Survey Results:

After completing the game, participants filled out the HARUS survey, which we split into manipulability (ease of use) and comprehensibility (understanding the game's instructions) categories.

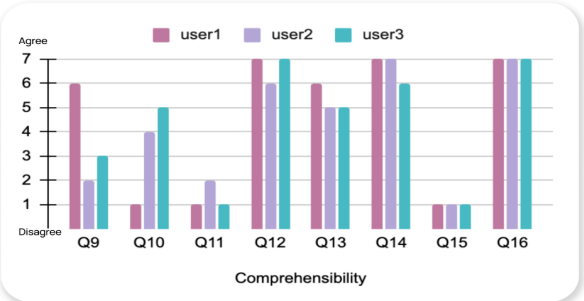
Manipulability statements

Q1	I think that interacting with this application requires a lot of body muscle effort.
Q2	I felt that using the application was comfortable for my arms and hands.
Q3	I found the device difficult to hold while operating the application.
Q4	I found it easy to input information through the application.
Q5	I felt that my arm or hand became tired after using the application.
Q6	I think the application is easy to control.
Q7	I felt that I was losing grip and dropping the device at some point.
Q8	I think the operation of the application is simple and uncomplicated.



Comprehensibility statements

Q9	I think that interacting with this application requires a lot of mental effort.
Q10	I thought the amount of information displayed on screen was appropriate.
Q11	I thought the information displayed on screen was difficult to read.
Q12	I felt that the information display was responding fast enough.
Q13	I thought that the information displayed on screen was confusing.
Q14	I thought the words and symbols on screen were easy to read.
Q15	I felt that the display was flickering too much.
Q16	I thought that the information displayed on screen was consistent.



Key findings included:

- **Manipulability**: Most users felt the game was easy to control, although one participant who spent a long time searching for the student card mentioned that the game felt physically demanding. Another participant felt the controls were somewhat difficult.
- **Comprehensibility**: The survey revealed that many participants were confused about the game's instructions, particularly regarding how to interact with the AR objects. This highlighted the need for clearer guidance and explanations within the game.

User Feedback:

During the follow-up interviews, participants shared valuable feedback on their experiences and provided suggestions for improvement:

1. UI Layout:
    - The map steps should be arranged in a bottom-to-top order, rather than upside down, for a more natural navigation flow.
    - A clear and consistent "Back to Map" or "Back to Previous Page" button should be present on all screens to prevent confusion.
    - The "Back to Map" button in Hint 1 was sometimes non-functional, which needs to be fixed immediately.
2. Interactive Experience:
    - Users expressed a desire for clearer instructions regarding how to use the phone's camera to activate the AR interactions.
    - Some were unsure if they had to stand up or move around to interact with the game, so it was important to clarify that staying seated and holding the phone is sufficient.
    - It was suggested that hint pictures be displayed on the first page of each mini-game for better context and understanding.
3. Function Optimization:
    - Provide more detailed explanations for the hints, showing exactly what users need to find in each stage.
    - Hint 2 needs significant improvement, as the surface detection was unreliable, and the student card was not clickable due to a bug.
    - The map page should include all essential game information to make it more intuitive.
    - Include more detailed instructions across all sections to avoid user confusion and improve overall comprehension of the game's mechanics.

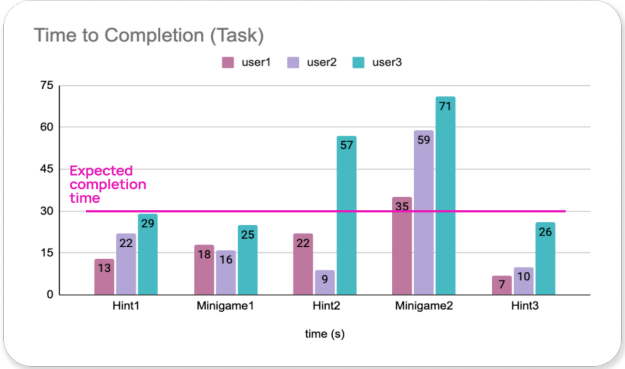
Adaptations Based on 1st Round User Testing:

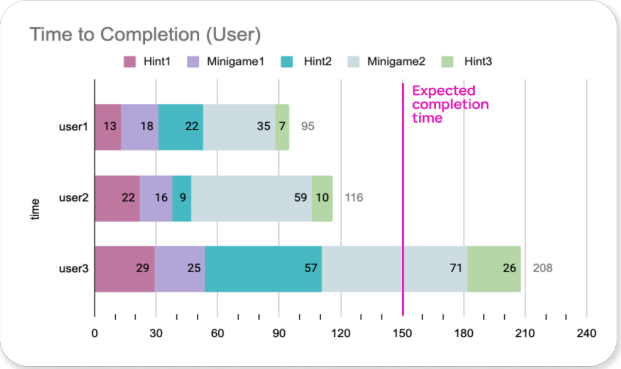
After the first round of user testing, we made some key adjustments to improve the overall user experience. We started by modifying the map navigation, starting steps from the bottom to help users navigate more effectively. To facilitate smoother navigation, we added consistent 'back to map' or 'back' buttons across all screens. We also improved the clarity of hints and instructions, specifying exactly what items users need to find and providing more detailed information in all sections to minimise confusion and guide users effectively. To increase engagement, background music (BGM) was introduced to create a more immersive experience, and vibration effects were added to enhance tactile feedback and enjoyment.

5.3.2 Second Round of User Testing

- **Test Scope**: The testing focused on Hint 1, Mini-Game 1, and Hint 2.
- **Participants**: Three THI students, all of whom are part of our target user group, participated in this round of testing.

Time to Completion:





In this round, the completion times aligned closely with our expectations (Hint 1, Mini-Game 1, Hint 2 and Hint 3 were all completed within 30 seconds). However, User 3 took longer than anticipated to locate the virtual item in Hint 2, indicating individual differences in AR interaction efficiency.

A key focus of this round was Mini-Game 2, a 3D puzzle introduced as a new feature. Observations revealed that all participants exceeded the 30-second estimate for this task, with User 3 taking the longest time. This highlighted areas for improvement in the game's design and instructions.

Breaks-in-Presence:

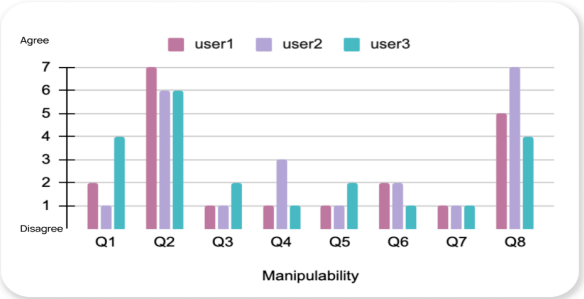
While the first three tasks were completed with minimal interruptions, several significant breaks occurred during Mini-Game 2. These breaks highlighted recurring issues:

- **QR Code Recognition:** All users experienced delays as the system struggled to recognise all four QR codes simultaneously. Many users tried to display all the QR codes in the camera's field of view at the same time. This caused further delays in recognition.
- **Physical Interaction:** Users found it difficult to move QR codes on the table with one hand while holding the phone with the other, resulting in slower progress.
- **Misaligned Expectations:** Users were naturally trying to drag AR puzzle pieces around the phone screen, similar to the drag-and-drop mechanic of Mini-Game 1. However,

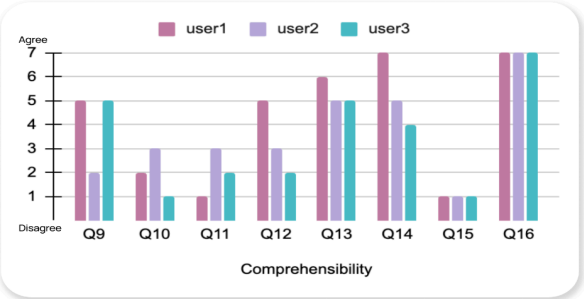
Mini-Game 2 required them to physically move QR codes on the table. This misinterpretation was due to insufficient guidance, causing confusion and delays.

HARUS Survey Results:

Manipulability statements	
Q1	I think that interacting with this application requires a lot of body muscle effort.
Q2	I felt that using the application was comfortable for my arms and hands.
Q3	I found the device difficult to hold while operating the application.
Q4	I found it easy to input information through the application.
Q5	I felt that my arm or hand became tired after using the application.
Q6	I think the application is easy to control.
Q7	I felt that I was losing grip and dropping the device at some point.
Q8	I think the operation of the application is simple and uncomplicated.



Comprehensibility statements	
Q9	I think that interacting with this application requires a lot of mental effort.
Q10	I thought the amount of information displayed on screen was appropriate.
Q11	I thought the information displayed on screen was difficult to read.
Q12	I felt that the information display was responding fast enough.
Q13	I thought that the information displayed on screen was confusing.
Q14	I thought the words and symbols on screen were easy to read.
Q15	I felt that the display was flickering too much.
Q16	I thought that the information displayed on screen was consistent.



- **Manipulability:** While the overall manipulability of the game met expectations, the difficulties in QR code recognition and physical interaction in Mini-Game 2 made users feel the game was harder to control than desired.
- **Comprehensibility:** Despite enhancements made after the first round (e.g., clearer hints and instructions), users still overlooked key instructions. There was ongoing confusion regarding the interaction methods in Mini-Game 2, particularly how it differed from previous tasks. This highlights a need for more explicit guidance and reinforced instructions to ensure players fully understand the mechanics.

User Feedback:

1. UI Layout
    - User 2 appreciated the addition of the "Back to Map" and "Information" buttons, noting that they improved navigation.
    - User 3 suggested not placing all four QR codes on the table initially to prevent confusion.
2. Interactive Experience
    - User 1 enjoyed the music and hints but suggested incorporating real-world locations for more immersive hints.

- User 2 wanted more sound effects (e.g., level-up sounds) and clearer QR code instructions.
- User 3 struggled with 360° movement in AR and felt frustrated due to skipped or unclear instructions. They suggested providing a clear mission overview at the start of each task and more specific hints for AR interactions.

3. Function Optimization
   - User 1 pointed out the need for clearer differentiation between the interaction styles of Game 1 and Game 2.
   - User 2 noted a bug where re-entering Mini-Game 2 from the map caused progress to reset.
   - User 3 requested more detailed guidance for using QR codes and completing the 3D puzzle in Mini-Game 2.

Adaptations Based on 2nd Round User Testing:

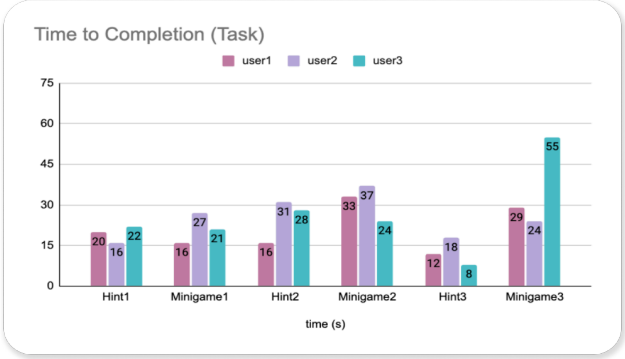
Following the second round of user testing, we focused on refining the interaction elements, particularly in Mini-Game 2. Clear prompts were added at the start and throughout the game to inform users how to interact with the QR codes, emphasising the physical movement of the QR code rather than dragging it across the screen. Detailed instructions on how to effectively display all QR codes were provided to avoid recognition delays. A brief mission overview was included before starting Mini-Game 2 to ensure users understood the task at hand, and more sound effects and cues were added to enhance immersion and feedback, creating a more engaging experience.

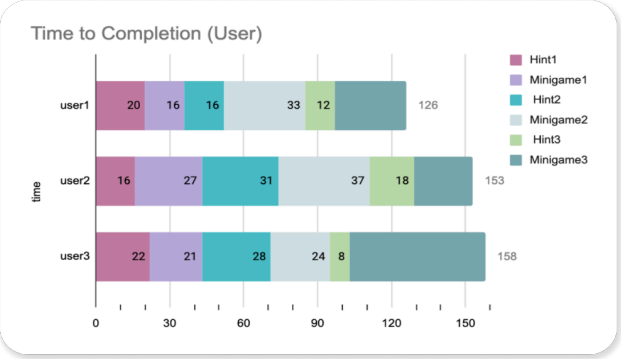
5.3.3 Final User Testing

- **Test Scope**: This round of testing focused on Mini-Game 3 and the overall game flow, particularly the transitions between the final tasks.
- **Participants**: Three THI students, all of whom are part of our target user group, participated in this round of testing.

Time to Completion:

Completion times of Mini-Game 3 varied significantly, ranging from 20 seconds to over a minute. Users who quickly grasped the timing mechanic completed the task efficiently, while others required multiple attempts due to challenges with timing and speed.





These results suggest that while the overall game flow is manageable, the speed of the moving Student ID Card and differences in user interaction styles contributed to discrepancies in completion times.

Breaks-in-Presence:

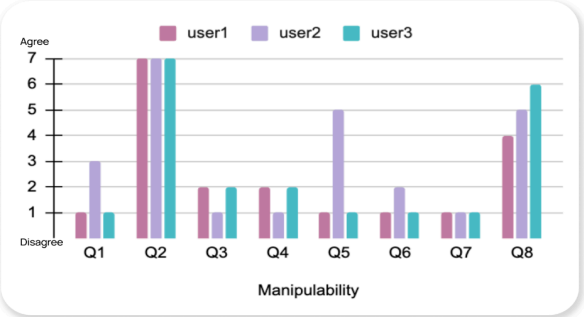
Several breaks-in-presence were observed during this round, most of which occurred in Mini-Game 3:

- **Timing Misalignment:** Users struggled with the timing mechanic of tapping the screen to align the card with the virtual printer. Three out of four participants missed the target on their first attempt, leading to frustration.
- **Speed of Moving Card:** All participants commented that the Student ID Card moved too quickly, making it difficult to tap precisely when it aligned with the printer. This caused repeated failures and increased frustration.
- **AR Environment Issues:** Participants had difficulty detecting the virtual Printer in environments with poor lighting or overly complex backgrounds. This caused minor interruptions as users repositioned their phones to better scan the area.

HARUS Survey Results:

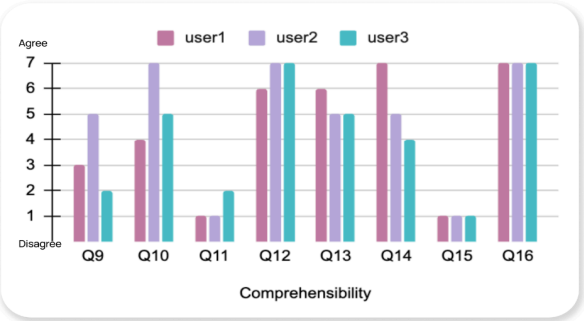
Manipulability statements

Q1	I think that interacting with this application requires a lot of body muscle effort.
Q2	I felt that using the application was comfortable for my arms and hands.
Q3	I found the device difficult to hold while operating the application.
Q4	I found it easy to input information through the application.
Q5	I felt that my arm or hand became tired after using the application.
Q6	I think the application is easy to control.
Q7	I felt that I was losing grip and dropping the device at some point.
Q8	I think the operation of the application is simple and uncomplicated.



Comprehensibility statements

Q9	I think that interacting with this application requires a lot of mental effort.
Q10	I thought the amount of information displayed on screen was appropriate.
Q11	I thought the information displayed on screen was difficult to read.
Q12	I felt that the information display was responding fast enough.
Q13	I thought that the information displayed on screen was confusing.
Q14	I thought the words and symbols on screen were easy to read.
Q15	I felt that the display was flickering too much.
Q16	I thought that the information displayed on screen was consistent.



- **Manipulability:** While participants appreciated the simplicity of the tap-to-align mechanic, the speed of the Student ID Card and the lack of visual aids made Mini-Game 3 feel less intuitive and more stressful.
- **Comprehensibility:** Instructions for Mini-Game 3 were mostly understood; however, the lack of guidance on timing precision and unclear hints about the moving speed of the card led to repeated failures. Participants suggested introducing visual cues (e.g., a progress bar or countdown) and reducing the speed of the moving card to improve comprehensibility and ease of interaction.

User Feedback:

1. UI Layout
    - User 3 requested a more obvious visual indicator, such as a blinking light or progress bar, to show when the card aligns with the printer.
2. Interactive Experience
    - User 1 liked how it all flowed together, but they found the AR detection for Hint 3 frustrating in dark places. They suggested adding a guide on the screen to help users position their phones correctly.
    - User 2 thought that the Student ID Card was too fast in Mini-Game 3, which made it feel more like a test of reflexes than a puzzle. They suggested slowing down the card or adding different difficulty levels to make it more accessible.

- User 3 had the same problem with the fast-moving card and suggested adding a practice round to help users get used to the timing mechanic before the actual challenge.

3. Function Optimization
   - User 1 found a small problem where the printer model would briefly disappear after it was detected, meaning the user had to rescan.
   - User 3 said that a tutorial or calibration step would be useful, so that players can get used to the mechanics before the actual game.

Adaptations Based on Final User Testing:

After analyzing feedback from the final round of user testing, we implemented several targeted improvements to address the most pressing issues identified by participants. First, we resolved a persistent bug where the printer model in Hint 3 would temporarily disappear after detection, ensuring a smoother AR experience and reducing the need for users to rescan the environment. In Mini-Game 3, the speed of the moving Student ID Card was reduced to provide users with more time to align the card with the printer, making the task more accessible and less frustrating. These changes were aimed at improving usability while maintaining the challenge and engagement of the gameplay.

6. Project Feasibility and Challenges

6.1 Feasibility Analysis

The development and implementation of “Discover THI” demonstrate significant potential in terms of technical, operational, and user engagement aspects, though some challenges and limitations were identified.

Technical Feasibility: The integration of AR mechanics and mini-games is achievable using existing technologies, including reliable AR libraries and mobile development platforms.

Operational Feasibility: The game’s structure, including the Map Page navigation and linear task progression, ensures a smooth operational flow. The modular design allows for easy iteration and expansion, while the clear division of hints and mini-games ensures manageable development milestones. Testing has shown that users can complete tasks within expected timeframes, demonstrating operational viability.

User Feasibility: The game has shown strong engagement potential among THI students, aligning with their interest in interactive and exploratory tasks. Feedback from user testing highlights areas for refinement, such as clearer instructions, adjusted difficulty, and enhanced AR guidance. These improvements are feasible within current development resources and timelines.

Limitations: Despite the overall feasibility, there are notable limitations. The reliance on mobile devices with specific capabilities (e.g., high-quality cameras for AR) may exclude some users. Consistent user feedback and iterative testing to address usability issues can extend development timelines and require ongoing resource allocation. Additionally, the game's current design focuses primarily on entertainment and engagement, and its educational impact could be further strengthened. Incorporating more educational content and aligning tasks with real-world academic or practical skills would enhance its value as an educational tool.

6.2 Challenges Encountered

The development of "Discover THI" presented a range of challenges across conceptualization, design, development, and testing phases. Each phase had its unique obstacles, which are detailed below:

Game Concept:

- **Concept Differentiation:** Initially, significant effort went into brainstorming a concept to help THI freshmen adapt to campus life. During discussions with peer students, we discovered that two other teams had similar ideas. To differentiate our project, we revised the user flow and adjusted the objectives and mechanics of Mini-Game 2 and Mini-Game 3. These iterations led to the final user flow and game structure, ensuring originality while maintaining relevance to the target audience.
- **Game 3 Concept Simplification:** The original plan for Game 3 involved detecting a real printer, but this proved too complex and resource-intensive. We simplified the concept by replacing printer detection with QR code detection, which was more feasible within our time frame.

Design Challenges:

- **UI and Accessibility:** Balancing the visibility of UI elements without overcrowding the screen was a persistent challenge. Ensuring accessibility required extra focus on contrast, text size, and visual clarity to make the game user-friendly for all players.

- **Touch Controls:** Designing touch-friendly controls with clear and responsive feedback mechanisms required iterative refinements. Switching to Unity's new input system to support touch controls on mobile devices added complexity and required extensive testing.

Development Challenges:

- **Learning Curve:** With only one team member possessing a programming background, the initial learning curve for Unity development and coding was steep. The team spent significant time self-learning Unity's workflow and scripting.
- **Debugging and Performance:** Debugging Game 2 was challenging due to frequent freezes. Prof. Markus introduced an extension that allowed us to print debug logs directly to a laptop, helping us identify and resolve an unhandled exception that was causing crashes.
- **Data Management:** Saving player profiles and progress in JSON format turned out to be more challenging than expected. Ensuring data consistency and proper functionality across sessions required careful handling and debugging.
- **3D Model Challenges:** Finding free, high-quality 3D models was more difficult than expected. We initially tried creating our own models using Blender, but this approach was time-consuming and required advanced skills. Given our limited time and resources, we simplified the game concept to something more manageable.
- **Prefab and Gameplay Issues:** During Game 3, we encountered prefab-related issues, including the student card disappearing when moved along the X-axis and prefabs overlapping or disappearing altogether when the printer mechanic was implemented. These problems disrupted gameplay and required significant troubleshooting. Additionally, implementing images as materials on prefabs caused unexpected issues, necessitating debugging and adjustments.

Testing Challenges:

- **Device Compatibility:** The team used Windows computers for Unity development, while testing required Android devices incompatible with their iPhones. A dedicated Android phone was purchased specifically for testing purposes. Using emulators also proved unreliable and difficult to set up.
- **Environment Challenges:** Working with a 3D environment posed additional difficulties, as it didn't provide the experience we initially envisioned. This required adjustments to align the gameplay with our technical capabilities.

Despite these challenges, all testing phases were successfully completed. The feedback gathered during testing allowed us to address critical issues and refine the game design, resulting in a more polished and engaging experience.

7. Project Reflection

7.1 Achievement of Goals

The "Discover THI" project successfully met the requirements by creating a mobile AR treasure hunt game that integrates hints and mini-games. The game was developed using Unity and AR Foundation, ensuring compatibility with modern smartphones and leveraging AR features like plane detection, object placement, and image tracking for an engaging user experience. Players navigated through the game by scanning their surroundings to find virtual objects, such as a backpack, student ID, and printer. These hints unlocked access to three interconnected mini-games, completing the treasure hunt progression.

Mini-game 1 was a 2D drag-and-drop puzzle that allowed players to arrange pieces on the screen to reveal a student ID card. This met the requirement for a 2D game without AR elements and provided a simple, intuitive introduction to the game mechanics. Mini-game 2 incorporated AR and image tracking through the use of physical QR codes. Players scanned the QR codes to reveal virtual 3D puzzle pieces and rearranged them to complete the task. This effectively utilized AR tracking technology while introducing a more interactive challenge. The final mini-game combined AR with action elements, where players scanned a QR code to locate a virtual printer and timed their taps to align a moving card with the printer. This action-oriented task added variety and excitement while fulfilling the requirement for an AR mini-game with dynamic interaction.

The theme of helping THI new students adapt to campus life was integrated into all aspects of the game. Tasks such as locating a backpack, finding a student ID, and simulating a printing task reflected familiar campus-related activities, blending education with entertainment. By aligning gameplay with the experiences of THI students, the game created a meaningful connection for the target audience.

7.2 Goal Changes

During the project, some goals evolved based on user feedback and feasibility. Initially, Mini-Game 2 was designed to allow users to complete a 3D puzzle directly on the screen. However, we decided to combine real-world actions with virtual tasks by having users scan physical QR codes and rearrange them to solve the puzzle. This change enhanced the integration of physical and virtual elements, making the experience more immersive. Similarly, for Mini-Game 3, the original plan was to have users scan a real printer and then align a virtual student ID card with a specific location on the screen-based printer. Due to time constraints, we instead chose to display a virtual

printer by scanning a QR code. This approach simplified development but reduced the potential for real-world educational relevance, which could be reconsidered in future iterations to better integrate physical interactions.

7.3 Future Iterations and Final Reflection

In a future iteration, several goals would need further refinement. Improving AR functionality in varying environmental conditions, such as low light or cluttered spaces, remains an essential focus. Enhancing instructions and feedback systems across all tasks would ensure players understand the mechanics more intuitively. Additionally, deeper integration of real-world THI campus locations and knowledge could further immerse users and strengthen the educational value of the game. Expanding accessibility features, such as customizable controls or options for visually impaired users, would also broaden the game's appeal.

Although we identified areas where our project fell short compared to others, we are proud of our final product. These challenges provided valuable learning opportunities, and we are committed to further refining the game. By addressing the identified issues and learning from peer projects, we aim to optimize the user experience and elevate the quality of "Discover THI."

8. Conclusion

8.1 Key Findings

The development and testing of "Discover THI" revealed several key insights into the game's design, user experience, and areas for refinement:

1. The linear task-based structure, guided by the Map Page, provided players with a clear and engaging progression. The visual representation of task statuses helped users stay oriented throughout the game, contributing to an intuitive game flow.
2. The AR mechanics, such as finding the backpack, student ID, and printer, successfully encouraged exploration and physical engagement. However, challenges with detection in low-light or cluttered environments impacted user immersion, indicating a need for further optimization.
3. Dense instructions and limited contextual feedback created user confusion during critical moments, such as AR scanning failures or missed taps. Clearer step-by-step guidance and

timely feedback were identified as essential improvements to enhance user understanding and satisfaction.

4. Background music, sound effects, and tactile feedback effectively enhanced the game's atmosphere, contributing to immersion and engagement. Incorporating more THI-specific locations or knowledge into tasks could further strengthen the connection to the real-world context.
5. Iterative testing was invaluable in refining the game. Feedback led to improvements such as optimized instructions, better AR functionality, and speed adjustments in Mini-Game 3, demonstrating the importance of user-centered design.

These findings highlight the strengths and opportunities of "Discover THI," laying the foundation for further enhancements to create a more seamless and immersive user experience.

8.2 Recommendations for Future Work

User Education, Feedback, and Help Information: We need more detailed and dynamic tutorials to help users understand and navigate the game. These could include text-based, animated, or video-based guides demonstrating key parts of the game. Tutorials should break down instructions into smaller, step-by-step instructions to avoid overwhelming users. The game could have in-app help features that step in when users are struggling. For example, if a user keeps failing at a task, a pop-up could offer tips or hints. During AR scanning tasks, the game could display alignment guides or troubleshooting tips when no objects are detected. These improvements would make the game more supportive.

Overall Game Flow: The tasks could be made smoother by making the transitions between hints and mini-games more seamless. Each task should start with a mission summary to make it clear what the objective is. This will help users to understand what is expected of them.

Map Page: The map page could have moving icons to show which tasks are completed, in progress, or newly unlocked. Introducing a dynamic difficulty adjustment system could improve the flow by adapting the challenge level to each user. For example, if a player struggles with a mini-game, the system could offer an easier version or additional assistance.

Hints (AR Elements): To ensure users can better understand and navigate the game, more detailed and dynamic tutorials should be introduced. These tutorials could include not only text-based instructions but also animated or video-based guides demonstrating key mechanics, such as AR scanning, puzzle assembly, and timing-based tasks. Instead of presenting large blocks of text, tutorials should break down instructions into manageable, step-by-step prompts that are

displayed progressively to avoid overwhelming users. By implementing these improvements, the game can foster a more supportive and frustration-free user experience.

Mini-Game 1: Improve feedback mechanisms by introducing animations or glowing effects to confirm when a puzzle piece is correctly placed. This would provide users with immediate, satisfying visual reinforcement and reduce confusion about their progress.

Mini-Game 2: Refine QR code recognition to eliminate delays and ensure the system can identify codes more reliably, even if only partially visible in the camera's frame. Additional visual prompts, such as arrows or icons, could illustrate how users should physically manipulate the QR codes to assemble the 3D object.

Mini-Game 3: Expand on recent improvements by adding a practice mode or calibration step before the task begins. This feature could allow users to familiarize themselves with the timing mechanics in a low-pressure setting. Additionally, dynamic difficulty adjustment could make the card move faster or slower based on the player's performance. Finally, distinct sound cues for successful or missed attempts would enhance feedback and immersion.

Real Educational Meaning: To create a stronger connection between the game and the real-world environment of THI, the game could incorporate tasks tied to actual campus locations. For example:

- Hint 1: Users could be required to visit specific areas on campus, such as the library or student lounge, to scan AR markers or locate virtual items.
- Hint 2 and Hint 3: Searching for the student ID card or printer could involve exploring real buildings or rooms within THI, adding an exploratory element to the gameplay.
- Additionally, mini-games or hints could incorporate trivia or information about THI, such as historical facts, notable alumni, or campus events. This integration of real-world content would make the game more immersive and educational, fostering a deeper connection between players and their university environment.

By addressing these areas, future iterations of the game can create a more intuitive, immersive, and engaging experience that not only entertains but also strengthens users' connection to the real-world context of THI.

9. Acknowledgments

We extend our heartfelt gratitude to our practical mentor, Markus Weißenberger, for his invaluable guidance throughout the Unity development process. His expertise and insights were crucial to the success of this project. We are also profoundly grateful to Professor Luisa Mayershofer for

attentively listening to each of our presentations and providing valuable feedback. We deeply appreciate the support and participation of our test users, who generously shared their time and experiences. Their feedback and suggestions were instrumental in refining and improving our design.

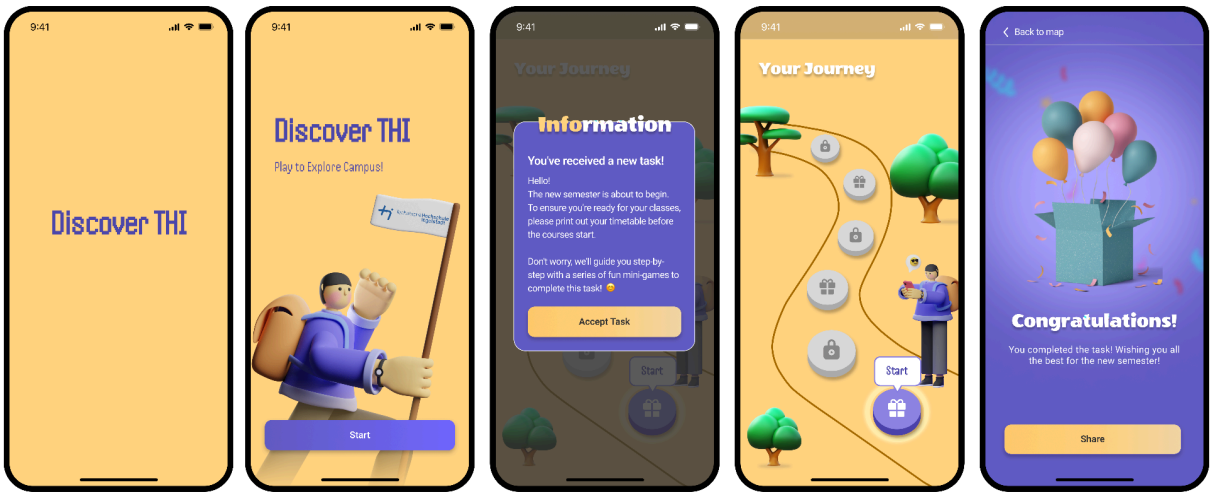
APPENDICES

Appendix A: Hi-Fi Design

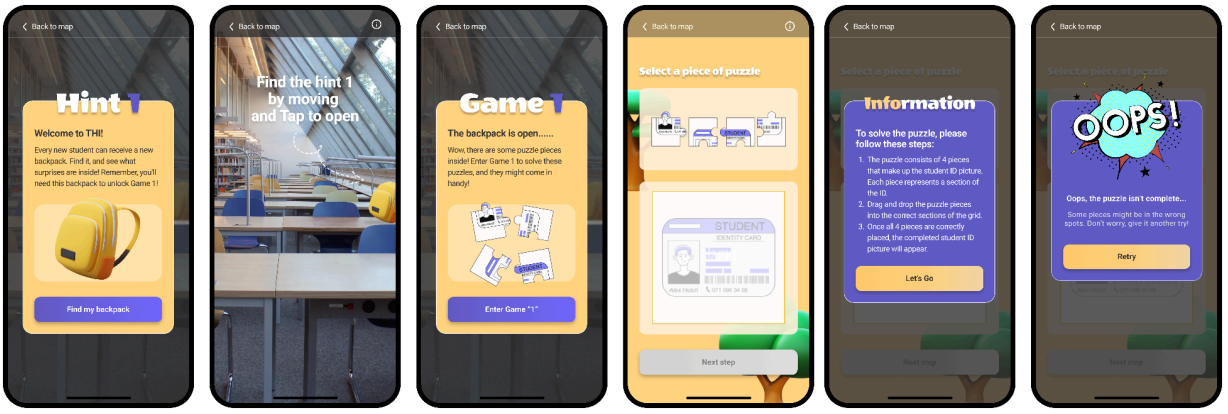
Figma Links :

https://www.figma.com/design/Djod9aPOLLhIO6nK1YzZjV/AR-Game-Design?node-id=631-5611&t=QPIG7UI4QImk3AM8-1

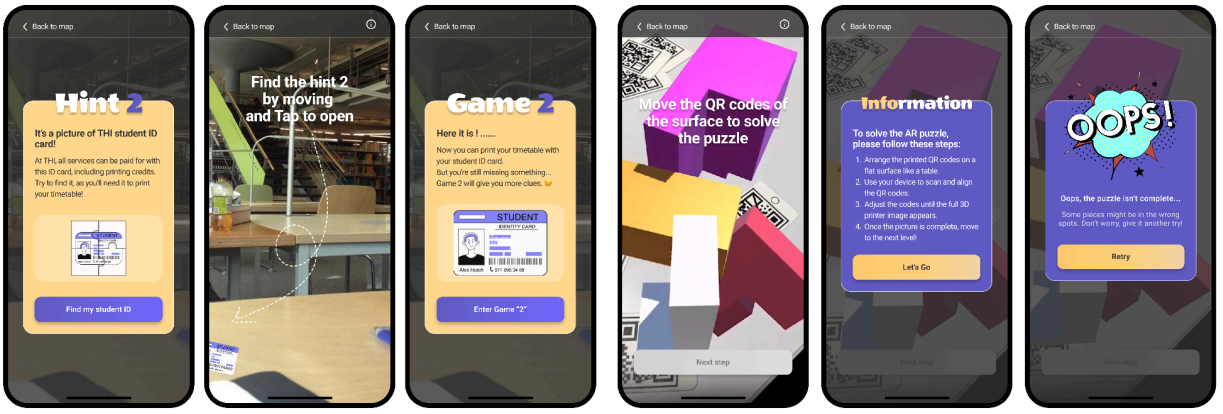
Game UI design:



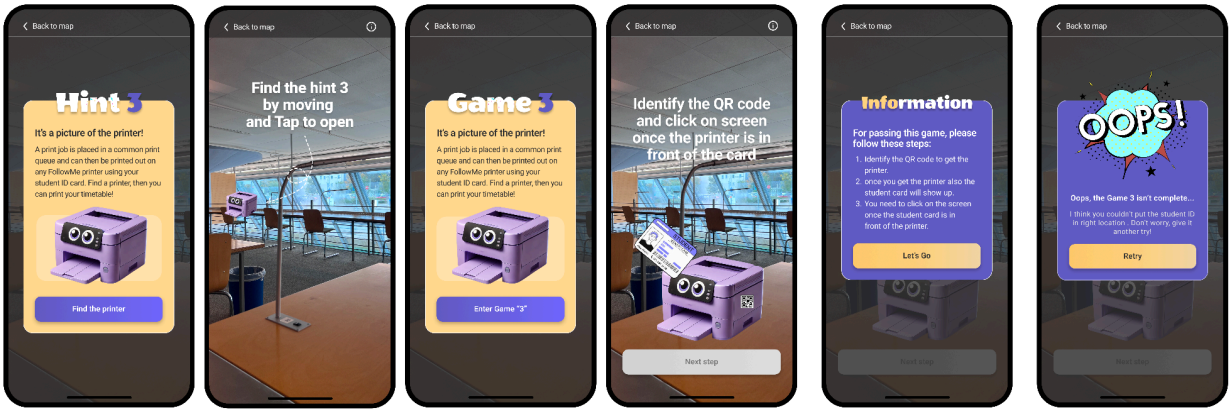
1. Loading Page / 2.Starting Page / 3.Information Starting Page / 4.Level Map / 5.Ending



1.Hint 1 pop up / 2.Hint 1 / 3.Game 1 pop up / 4.Game 1 / 5.Information for game 1 / 6. Error



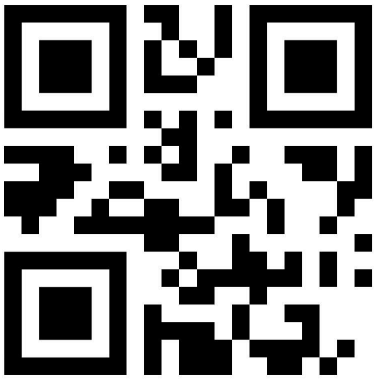
1.Hint 2 pop up / 2.Hint 2 / 3.Game 2 pop up / 4.Game 2 / 5.Information for game 2 / 6. Error



1.Hint 3 pop up / 2.Hint 3 / 3.Game 3 pop up / 4.Game 3 / 5.Information for game 3 / 6. Error

Appendix B: QR Codes for Minigame 2

**Must be in this order to make the full, correct shape.



Part 1

Part 2

Part 3

Part 4

Appendix C: QR Code for Minigame 3