

# TimeTraveler: The Lost Key

**Varshini, Akshaya, Chandana, Mani Harshitha, Yaaghnetha**

September 29, 2024

# PROJECT DESCRIPTION

- A time travel adventure game where the player must find a lost key to return to the present.
- The player's time machine malfunctions, causing them to land in an unknown timeline.
- The player must navigate through various historical or futuristic settings to find the key.
- After completing certain levels and get the key, the player will return to the present and completes the game.

## FEATURES

- **Player Movement:** Navigation of Character and Camera through arrow keys, Collision Detection and Rotation.
- **3D Environment:** Maze generation using Recursive Backtracking Algorithm and conversion of 2D map to 3D.
- **3D Environment:** Use of realistic wall and floor textures and skybox integration
- **Minimap:** The minimap updates in real-time to reflect the player's current position within the maze.
- **Minimap:** The minimap has an end-marker, and reflects the maze's layout accurately.
- **Camera:** The third-person camera provides an follow up to the character making the user to easily control the character.
- **Audio Integration:** Sound effects are triggered when the player collides with walls or reaches the end of the maze.
- **User Interface:** Addition of loading screen and responsive design makes it easy for different device widths.

# Tech Stack

- **Frontend:** Three.js, WebGL, HTML Canvas
- **3D Character Modelling and Designs:** Blender, Clara.io, SketchFab, Playground
- **Sound:** Three.js Audio API

## CHALLENGES FACED

- **Collision Detection:** Used bounding boxes(THREE.Box3), optimized collision checks.
- **Character loading:** Had issues with rendering the scene before models/textures fully load.
- **Minimap generation:** Integrating 2D elements like minimaps with the 3D scene was more challenging than rendering 3D scene.
- **3D Map implementation:** Mapping 2D maze to 3D walls.
- **Sound Effects and Character handling:** Adding sounds and modelling the character.
- **Project Management:** Managing timeline of all the team members.
- **Error Handling and Debugging:** Used try-catch blocks, logged errors, debugged with browser tools.

# Development Progress

- **Project Planning: June 28-July 05:**

- Conceptualized the game idea and set project goals.
- Established the timeline whom have to work for which part of the project.

- **Design Phase: July 09-July 13:**

- Designed the 3D maze layout, homepage and selected textures.
- Planned the user interface, including the minimap.

- **Implementation: July 13-Aug 10:**

- Set up the Three.js scene and added lighting.
- Implemented player movement, collision detection, and minimap integration.

- **Optimization: Aug 10-Aug 12:**

- Minimap Integration, Character Loading.
- Improved collision detection for better performance.

- **Testing and Debugging: Aug 13-Aug 16:**

- Testing and fixed bugs.
- Ensured responsive User Interface.

- **Final Adjustments:**
  - Polished visuals and refined gameplay mechanics and character controls.
- **Documentation:**
  - Documented code and created user guides.
  - Compiled a project report summarizing the development process.
- **Tools and Technologies:**
  - Utilized **Three.js**, **WebGL**, and **JavaScript** for development.
  - Used **Clara.io**, **Blender**, **SketchFab**, **Adobe Firefly**, **Playground** for Character design and modelling, Key, Environment.
  - Managed version control and collaboration using **Git**.

## OUTCOMES

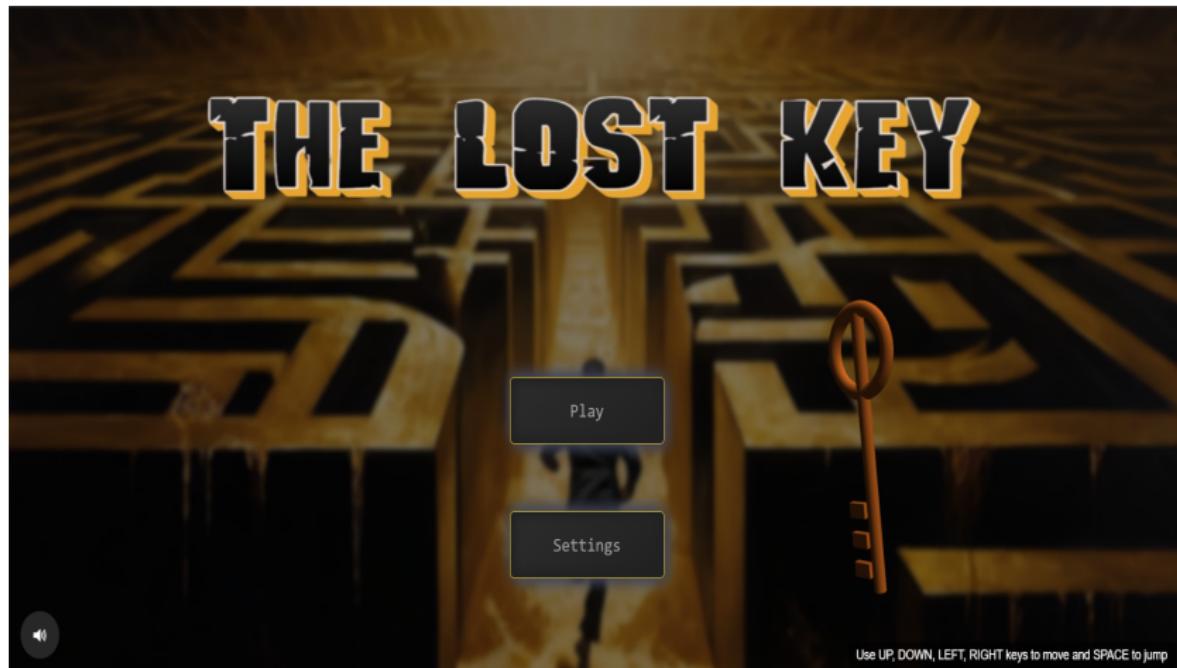


Figure: HomePage

# OUTCOMES



Figure: LEVEL 1

# OUTCOMES



Figure: LEVEL 2

# OUTCOMES

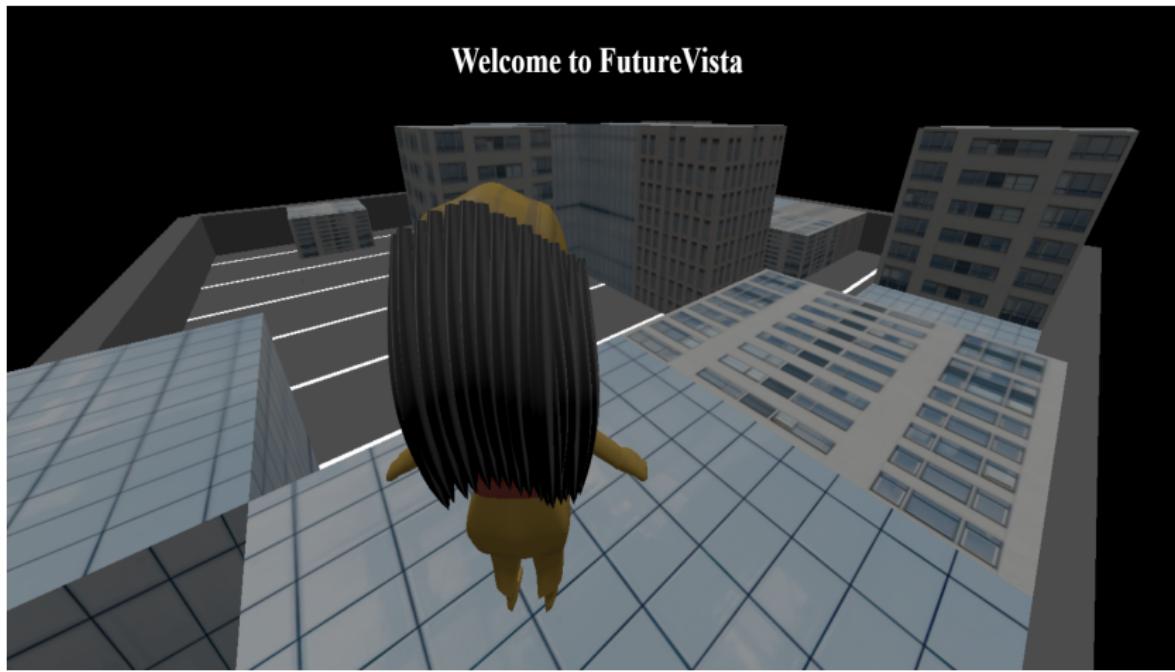


Figure: LEVEL 3

# OUTCOMES

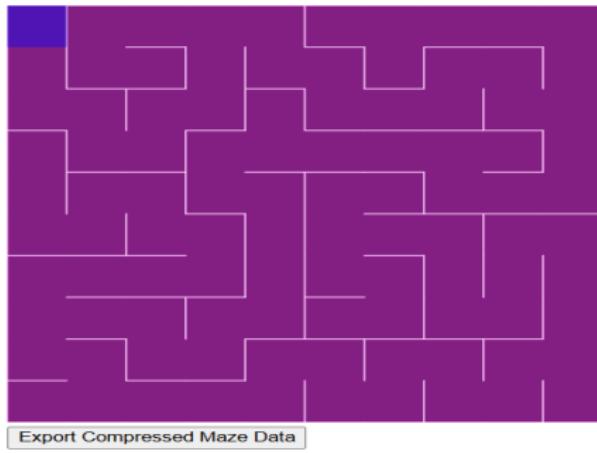


Figure: MAZE

# INSPIRATIONS

- <https://summer-afternoon.vlucendo.com/>
- <https://github.com/demonixis/Maze3D>

# LESSONS AND SKILLS LEARNED

- **Technical Skills:**

- Learnt a lot in Three.js, WebGL, and JavaScript for 3D game development.
- Improved understanding of 3D graphics, including lighting, shading, and camera controls.
- Debugging complex issues, such as collision detection and asynchronous loading.

- **Collaboration and Communication:**

- Enhanced collaboration skills through version control and teamwork.
- Improved communication by documenting the project and presenting it effectively.

- **Adaptability:**

- Developed the ability to quickly learn and adapt to new tools and technologies, here a completely new library.
- Improved flexibility in problem-solving by trying different approaches to achieve planned results.

- **Future Improvements:**

- Multiplayer options and leaderboard integration integrating backend.
- Addition of more number of levels with different environments.

# CONTRIBUTION TO THE PROJECT

- **HomePage:** Akshaya - Complete

- Included Play, Quit, Sound buttons, and Instructions. 3D Key Integration and Linking of HomePage to Level 1.

- **Level 1:** Chandana, Varshini

- Description: The player is stuck in a cave, where he has to reach the end point with the help of minimap.
  - Path Generation: Random path generation.
  - Chandana: Sound, Player Movements, Collision Detection.
  - Varshini: Maze and Minimap Generation, Setting up of scene and environment, End Point Detection, Linking of level 1 to 2 and player loading.

- **Level 2:** Yaaghnetha - Complete

- Description: The player is stuck in an ancient maze and has to reach the end point from his current position with the help of minimap.
  - Maze Generation: Generated 2D maze and integrated the maze data to make a 3D model of it.
  - Setting up of scene which includes, minimap, camera, floor and wall textures, skybox integration and player loading, Linking of level 2 to 3.
  - Inclusion of physics: Collision Detection, Player Movements, End Point Detection.

- **Level 3:** Varshini - Complete

- Description: The player is now stuck in a futuristic era and has to jump through the buildings and get the final key to complete the game(no help of minimap).
- Setting up of scene which includes, camera, floor and building textures, and player loading, Linking of level 3 to HomePage.
- Inclusion of physics: Collision Detection, Player Movements, End Point Detection.

- **Character:** Mani Harshittha

- Designed the game character from scratch using Blender.

- **Sound:** Chandana

- Integrated sound during collision detection for all the three levels.

*THANK YOU*