IT PAT Specs Document
Lost

Kian Anderson

Table of Contents

1.   Summary

Lost is a level based game, based loosely on the game Badlands, where the main character has been forced into a strange underground cave system in which he must escape to get back home. The game will start with a cinematic explaining to the player how he got there and then the player will play the first level, the levels get progressively harder as the player completes them. There is a currency with which the player can use to purchase different characters in the in game store. The player will be rewarded with a set amount of currency for every level they complete and also can find hidden bonus currency in the various levels in hidden chests.

Each level will have numerous obstacles that can kill the player once they touch it, this will be achieved by a collision detector that will instantly kill the character, a pop up message will appear to allow the user to respawn at the last checkpoint or the initial spawn point. Each level will have checkpoints that will allow the player to respawn at that point if they either get lost or die.

The characters will have a walking animation, a jumping animation and a resting animation. This will be achieved by repetitively cycling a sequence of pictures in the character which will create the effect of the character jumping and walking.

There will be a timer for each level, this indicate the amount of time the player took to complete the level, this data will be stored on a database if the previous time is greater, therefore storing the least amount of time taken for that specific level. The database will also store a player ID, their password and some help in playing the game.

The character will be controlled by the arrow keys and the AD keys, these keys will make the character either to the right or left, the player will not have control over vertical movement besides the jump function, the vertical height of the character will be determined by the ground underneath the character, the character will not be allowed to go below the floor and this will be achieved by using a collision detector, the jump movement will be controlled by the space button, this will launch the character into the air for a short time before it falls down to the floor.

The character will walk along an uneven floor, this will be achieved by making a simple floor layout that will use object collision detection to detect if the player is on the floor and move him to where the floor starts, the will also be various obstacles (such as spikes, spinning razor blades, etc.) which will also utilize object collision detection so that if the player happens to touch one of these objects, the player will die and return back to the east checkpoint or if that is not applicable, then the player will return back to the initial spawn point.

A tutorial in the from of a video will be available for the user to pay from the Help Screen. The video will explain the mechanics of the game and show the user the controls used to make the character move.

2.   Project Specifications

2.1 Game Rules

- The character will spawn at the spawning point on each level when the player starts a new level
- The player will move the character with the 'A' key to move left, the 'D' key to move right and the space key to jump
- The character must find their way to a cave exit to complete the level.
- The player may not play a level unless the level before that level has been completed previously.
- If the player dies, they will respawn at the last checkpoint, or if that is not applicable, then they will respawn at the start of the level.
- Every time the player completes a level they will earn 100 in-game currency.
- Every chest the player finds in a level they will be rewarded with 50 in-game currency.
- If the player jumps onto a platform they will be able to stand and walk on the platform.
- If the player reaches the finish point (marked by a green banner) they will finish the level and be taken to the Level Select Screen.
- When a player completes a level they will unlock the next level, that will be more challenging than the previous level, this will be stored for the next time they log in.
- If the player touches an obstacle they will die and be transported to the start of the level.

2.2 Program Functions Screen by Screen

2.2.1   Log in Screen

- The user is presented with a text field where their username is put in and another text field where the user must enter their password
- once the fields are filled out the user presses the login button and the username and password will be checked against the database to see if they are legitimate
- If it is the users first time logging in, there is a Sign Up button that the user must press that will allow the user to enter a username and password with a confirm password field that will add the user to the database
- Once one of these tasks have been completed the user will be taken to the Level Select Screen.

2.2.2   Help Screen

- The user will be able to access this screen from the Main Screen
- This screen will have different help tips that the user can select, depending on what the user has selected
- The information will be pulled from the database and displayed on the screen for the user to read the relevant information
- The screen will have a close button that will close the screen and return the player to the level select screen.

2.2.3   Best Time Screen

- This screen will be accessible through the Main Screen by use of a button
- This screen will display the players best time achieved on every level in a table format that can be easily read by the user
- The data will be pulled from the from the user class which is pulled from the database.
- The screen will have a close button that will close the screen and return the player to the level select screen.
- The screen will display the best time that has been achieved on the whole database by any user and compare it to the current player.

2.2.4   Game Screen

- The Game Screen will consist of the player the is centered in the middle of the screen with the background and the chosen level foreground and floor.
- It will consist of a pause button which will bring up a lead button to quit the level, as well as pausing the game
- The pause menu will also consist of a volume scroller to adjust the audio level and a mute button.
- There will be a floor designed specifically for each level that will utilize collision detection, this will slow for the character to not fall below the cave floor
- The character will spawn at the beginning of the level in a cave when they first start.
- If the player touches an obstacle, they will die.
- When a player collides with a platform the player jumping and falling will be set to false and the character will appear to walk in the platform.
- To finish the level the player must reach the finish line.
- If the player reaches a chest during the level 50 currency will be added to the database and if the player reaches the finish line 100 currency will be added to their database under the user name.
- If the player dies there will be a timer of 1 second after the player touches the obstacle which will add to the user experience and allow the player to recognize that they have died.
- A notification will be displayed "You died" when the player touches and obstacle.
- The user will have to complete the previous level to unlock the next level, when the user logs in, their level progress will be saved.
- A timer will begin as the player makes their first movement, this will only end when the player finishes the level.
- Depending on the level the user selects, the positioning of the obstacles and platforms will change to make the level more challenging.

2.2.5   Level Select Screen

- This screen will allow the player to select a level, but the user must have completed the previous level to access the level
- Otherwise it will be displayed as locked and the player will no be able to select it.
- When a level is selected a runnable thread is initialized in that level and the level screen is set to visible.

2.2.6   Main Screen

- This screen will consist of a button to allow the user to go to the level select screen
- A slider to adjust the volume of the game
- A button to get to the Level Select Screen
- A button to get to the Help Screen to display help.
- A button that will open a screen that will display the player with their best times achieved on every level and the best time that has been achieved on the whole database by any user.

3.   Specifications of User Interface

3.1   Theme for All Screens
- Each screen will utilize the Consolas font 'Lost'.
- Each screen will consist of a silhouette with bright vibrant colours as the background acting as horizons.
- The text will be white as the background will be mostly black due to the silhouette effect.
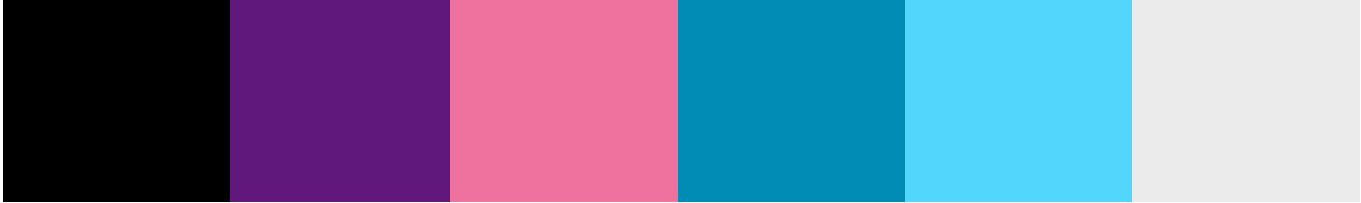
3.2 Screen Layout

3.2.1   Log in Screen

- The login screen will consist of two text fields that allow the player to input a username and a password in the centre of the screen
- There will be a label that says "Login"
- Two buttons below the text fields "Login" and "Sign Up"
- If the user presses the "Sign Up" button a third text field will appear below the original two that will allows the user to confirm their password.
- The background will be black and the text will be white on this screen.

3.2.2   Help Screen

- The Help Screen will consist of a label that says "Help" at the top of the screen, there will be a scroll function that will allow the user to select which help topic they would like to be displayed.
- The relevant help information will be displayed in a text area on the right side of the screen.
- There will be a button that will play the video tutorial.

3.2.3   Best Time Screen

- The High Score Screen will have a later that says "Best Time" at the top centre of the screen
- The best times will be displayed in a a text area that will be formatted with the level name, followed by the actual best time achieved by the player.

3.2.4   Game Screen

- The Game screen will have a silhouette that will act as the foreground and where the character can stand on
- The background will be a colourful horizon that the colour will vary depending on the level
- The character will be situated in the centre of the screen and the camera shall keep the character focused in the centre of the screen.
- The positioning of the character will vary depending on the level the user selects.

4.   Specifications of Help

The help screen will be accessible every screen through the use of a button in the top right of the screen, this will take the player to a separate screen where the user will be able to access the relevant topics which they would like to see helpful information on. A tutorial video will also be available from the help screen that will explain the game to the user.

4.1 Help Topics

- How to check High Scores
- How to play the game
- How to login as a new user
- Game Controls
- How to Sign In
- Different Levels in the game
- Basic Mechanics of the Game
- How to access the Tutorial Video
- Objective of the Game
- How to complete a level

4.1 Tutorial Video

- Show the user how to play the game
- Show the user the controls of the game
- Show the user the objective of the game.

5.   Specifications of Data Storage

5.1   Help.txt

Fields: Help Topic, Help Information
- These records are created when the program is created by the developer
- These records are accessed when a user goes into the help screen and selects a help topic
- These records are never updated

5.2   Users.txt

Fields: Username, Password, Level
- These records are created when a user selects Sign Up and enters their credentials
- These records are accessed when a player logs in, and when the player goes into the Best Time Screen
- These records are updated when a player gets a New Best Time in a level.

5.3   Levels.txt

Fields: Obstacle and Platform locations
- These records are created when the program is created by the developer
- These records are accessed when a user selects a level and the game screen is loaded.
- These records are never updated.

6.   System Requirements

6.1 User

6.1.1   Software

Operating System: Windows Vista or a later Version	To manage computer resources and run applications currently in use
Java JRE Version 7 or above	To decode and interpret Java bytecode.

6.1.2   Hardware

2.0 GHZ CPU (eg. Intel® Core™ 2 Duo E6600 AMD equivalent) or above	To process and run computer applications.
2 GB DDR3 RAM or more	To store the application currently in use.
- 2GB of storage (HDD or SSD) or more	To permanently store the application on the local hard drive.

6.2 Programmer

6.2.1   Software

Operating System: Windows Vista or a later Version	To manage computer resources and run applications currently in use.
NetBeans IDE 8.0.2	To provide and IDE for the programmer to code in.
Java JDK 7 or above (includes JRE to test and run the game)	To compile Java code and interpret Java bytecode

6.2.2   Hardware

3.0 GHZ CPU (eg. Intel i3 3700 or AMD equivalent) or above	To process and run computer applications..
4 GB DDR3 RAM or more	To store the application currently in use..
20GB of storage (HDD or SSD) or more	To permanently store the application on the local hard drive.

Bibliography