Name: Luke Zawis Mark _____________________/50

1. Brief introduction __/3

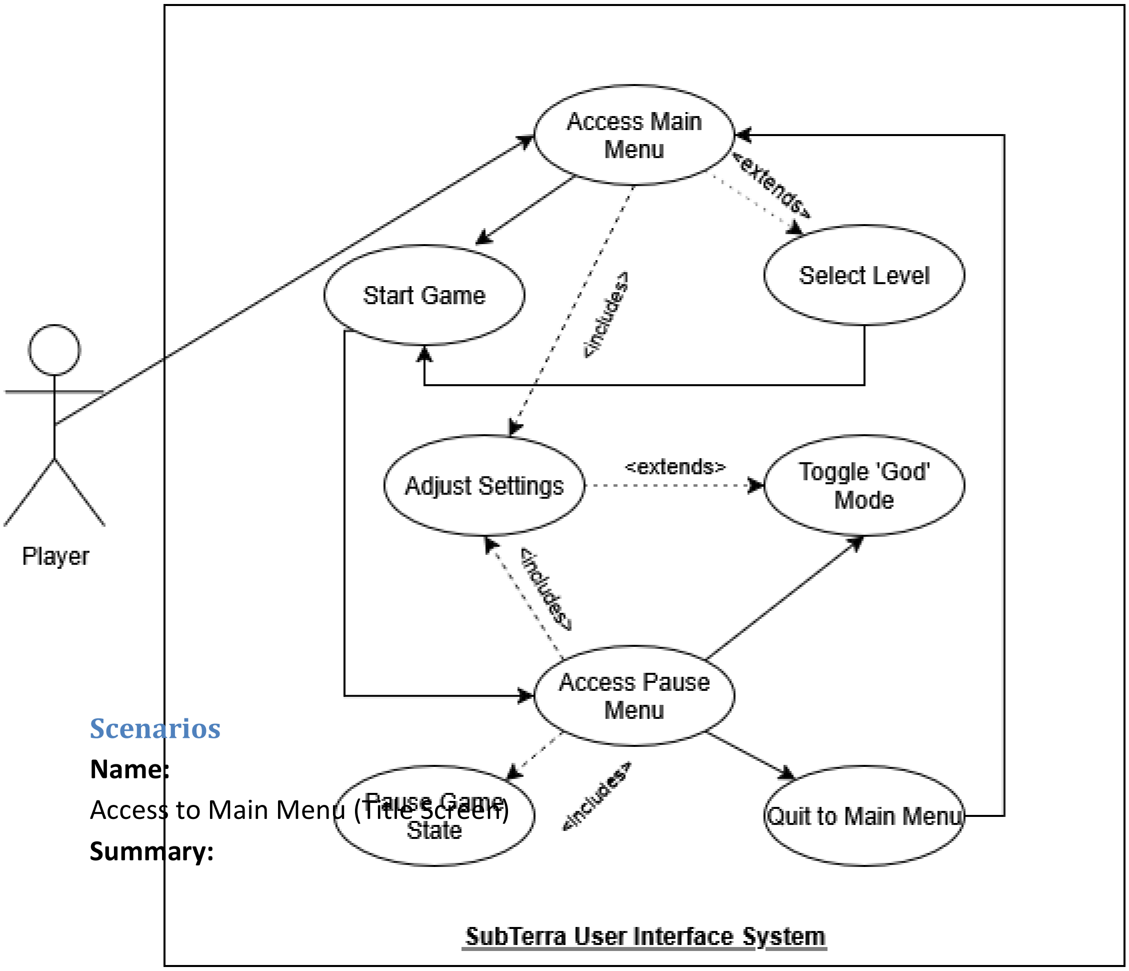
My feature for our videogame, SubTerra, will be creating the user interface which includes the main menu, pause menu, and other configurable settings the user can utilize.

When the user downloads and runs the game, my responsibility is to ensure that the first thing they see is a functional but also aesthetically pleasing title screen. This will include buttons that will each require an individual script. The title screen will allow the user to start the game, choose the level they wish to play, and finally miscellaneous settings such as sound control and a 'god mode' for testing purposes.

When in game, the user will also have access to a pause menu at all times. This menu will allow the user to configure settings such as sound volume as well as the ability to quit the current level and return to the title screen. The pause feature will also ensure that the game objects, player, and enemies are fully motionless or 'frozen' while the pause menu is active.

2. Use case diagram with scenario __14

Use Case Diagrams



Scenarios

Name:

Access to Main Menu (Title Screen)

Summary:

The user launches the game and interacts with the title screen to navigate options.

Actors:

User

Preconditions:

Game is launched by user and initialized.

Basic sequence:

1. Display title screen with buttons (start game, level select, settings, exit).
2. User selects desired button.
3. Process user selected action.

Exceptions:

Invalid input which will be ignored.

Post conditions:

User navigates to selected feature or remains in the main menu.

Priority: 1

ID: UI01

Name:

Start Game

Summary:

Gameplay is initiated by user in the main menu.

Actors:

User

Preconditions:

Main menu is displayed.

Basic sequence:

1. ‘Start Game’ button is pressed by user.
2. Load default level or last selected level.
3. Transition to in-game state.

Exceptions:

- No levels are available to select. (first playthrough)

Post conditions:

Gameplay begins at loaded level.

Priority: 1

ID: UI02

Name:

Select Level

Summary:

When prompted, the user chooses a level from the ‘Select Level’ menu on the main title screen.

Actors:

User

Preconditions:

Main menu displayed with accessible levels after ‘Select Level’.

Basic sequence:

1. ‘Select Level’ button is pressed by user.
2. List of unlocked levels is displayed.
3. User selects level.
4. Level is loaded and gameplay commences.

Exceptions:

- Invalid/locked level is selected
    - Error displayed and user is returned to level selection.

Post conditions:

Selected level is loaded and gameplay begins.

Priority: 2

ID: UI03

Name:

Access Pause Menu

Summary:

User pauses gameplay to access a pause menu screen.

Actors:

User

Preconditions:

Game is initialized and user is commencing in gameplay.

Basic sequence:

1. Uses presses pause key/button. (will vary by platform)
2. Freeze game state which includes player, enemies, and applicable objects.
3. Pause menu is displayed with options: ‘Resume’, ‘Settings’, and ‘Quit to Main Menu’
4. Process user selection.

Exceptions:

- User presses pause key while already paused. (only applicable to platforms that use a controller or keyboard)
    - Input is ignored.
    - Game will not resume until the appropriate button is pressed.

Post conditions:

Game is paused, resumed, or quits to the main menu. Settings can also be adjusted.

Priority: 1

ID: UI04

Name:

Adjust Settings

Summary:

The user is able to modify settings such as sound volume and ‘god mode’.

Actors:

User

Preconditions:

Main menu or pause menu is currently displayed.

Basic sequence:

1. User selects ‘Settings’ button within the main menu or pause menu.
2. Settings panel is displayed. (sound slider and ‘god mode’ toggle)
3. User adjusts settings
4. Settings are applied and user returns to the previous menu.

Exceptions:

- Invalid adjustment for sound volume. (<0 or >100)
    - Slider range is limited from 0 – 100 percent.

Post conditions:

Settings are appropriately updated and globally applied for the duration of the user session.

Priority: 2

ID: UI95

Name:

Toggle ‘God Mode’

Summary:

The user enables/disables ‘god mode’ for testing purposes. Player health and stamina become unlimited, and all available in-game items are added to the user’s inventory. Health and stamina values are then reset to defaults when the feature is disabled. When disabled, the user also loses all items within their inventory.

Actors:

User

Preconditions:

Settings menu is currently displayed.

Basic sequence:

1. User toggles ‘God Mode’ switch.
2. Unlimited health and stamina if enabled. User also receives all in game items when enabled.
3. Health and stamina are reverted back to default values when disabled. All items in the player’s inventory are removed when the mode is disabled.

Exceptions:

- God mode is unavailable as game is still in development.
    - Toggle is not visible to user in this state.

Post conditions:

God mode status is updated according to sequence listed above.

Priority:

3

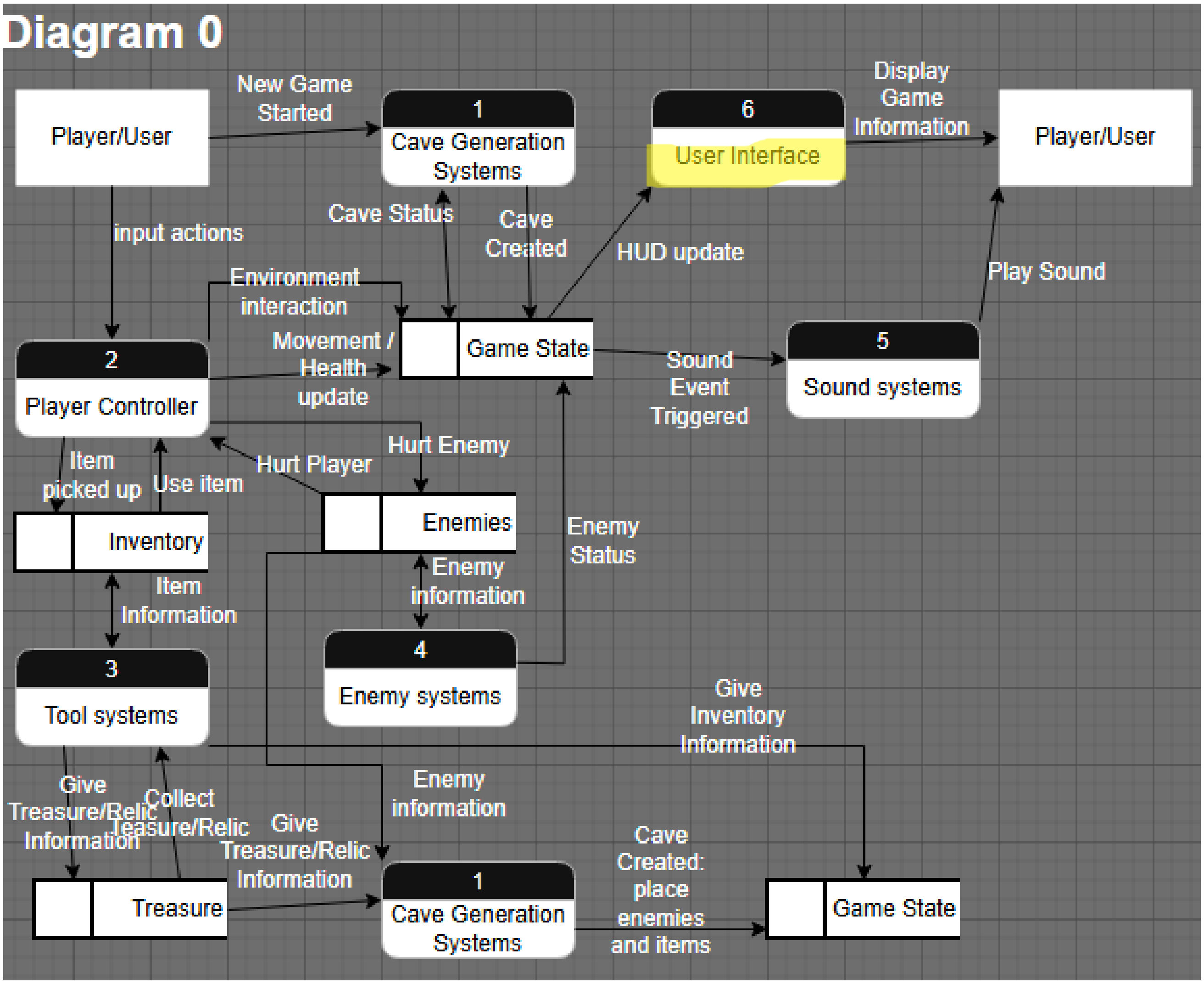
ID:

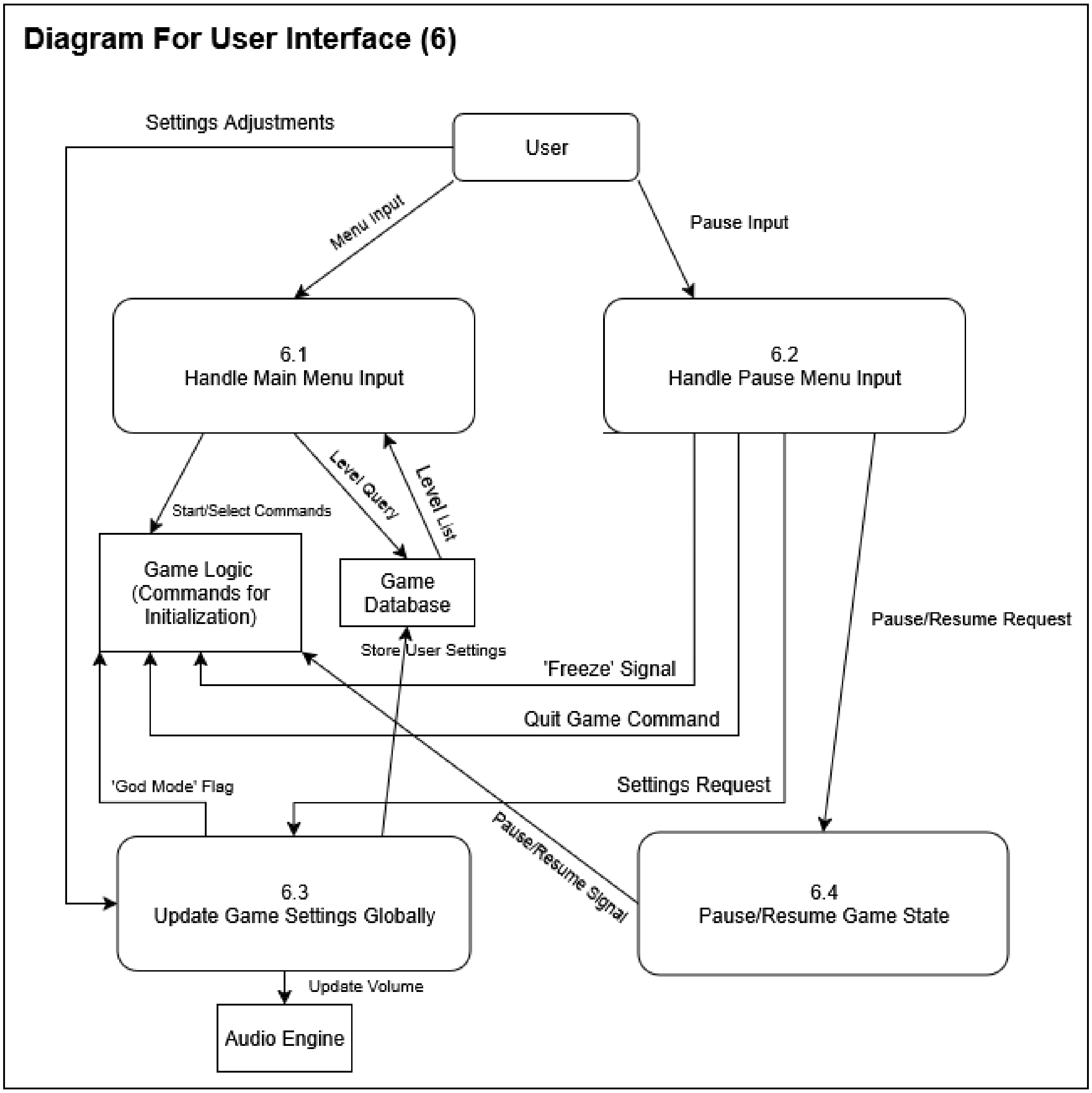
UI06

3. Data Flow diagram(s) from Level 0 to process description for your feature ______14

Within the diagrams below, the UI features will be covered.

Data Flow Diagrams





Process Descriptions

Handle Main Menu Input (6.1)

```
WHILE game is on main menu
         READ user input (click or key press)
         IF input = "Start Game"
           SEND start command to Game Logic
           Transition to gameplay
         ELSE IF input = "Select Level"
           READ level list from Game Database
           DISPLAY level list
           READ user level selection
           IF level is valid and unlocked
             SEND level selection to Game Logic
             Transition to gameplay
           ELSE
             DISPLAY error message ("Level locked")
             LOOP to level list
         ELSE IF input = "Settings"
           CALL Update Settings (6.3)
         ELSE IF input = "Quit"
           Terminate game
         ELSE
           IGNORE invalid input
       END WHILE
```

Handle Pause Menu Input (6.2)

```
WHILE in gameplay
    IF pause key (ESC) pressed
        IF not already paused
            SEND freeze signal to Game Logic
            DISPLAY pause menu (Resume, Settings, Quit)
            READ user selection
            IF selection = "Resume"
                SEND resume signal to Game Logic
                CLOSE pause menu
            ELSE IF selection = "Settings"
                CALL Update Settings (6.3)
            ELSE IF selection = "Quit to Main"
                SEND quit command to Game Logic
                RETURN to main menu
            ELSE
                IGNORE invalid input
        ELSE
            IGNORE (game is already paused)
END WHILE
```

Update Settings (6.3)

```
DISPLAY settings panel (volume slider, god mode toggle)
READ user adjustments
        IF volume adjusted
          VALIDATE volume in range 0-100
          IF volume < 0
            SET volume = 0
          ELSE IF volume > 100
            SET volume = 100
          SEND updated volume to Audio Engine
          SAVE volume to Settings Store
        IF god mode toggled
          IF enabled
            SET invincibility flag in Game Logic
          ELSE
            CLEAR invincibility flag
          SAVE god mode state to Settings Store
RETURN to previous menu
```

Pause/Resume Game State (6.4)

```
IF pause requested
   SEND freeze signal to Game Logic
   HALT updates for player, enemies, physics
ELSE IF resume requested
   SEND resume signal to Game Logic
   RESTORE updates for player, enemies, physics
```

4. Acceptance Tests ______9

These tests are designed to be fully automated and relies on no user input. Inputs are provided programmatically, and outputs are verified through system state checks. Each test is commenced 100 times to ensure stability.

Test Case 1: Main Menu Navigation/Functionality

Verify that the main menu process programmatic inputs correctly.

Inputs:

1. Launch game using test script
2. Send command to initiate gameplay ('Start Game' button is selected).
3. Send command to choose level 1.
4. Send command to adjust game volume from the MAIN menu.

Expected Outputs:

1. Game initiates.
2. Gameplay commences to default level.
3. Level 1 is selected.
4. Game is paused and volume is changed to a value of 50%

Test 2: Pause Menu Navigation/Functionality

Verify that the pause menu ‘freezes’ the game (player, enemies, objects), processes input, and resumes when prompted ‘unfreezing’ the game.

Inputs:

1. Send command to stimulate pause event.
2. Send command to resume gameplay.
3. Send command to trigger pause again but quitting to main menu instead of resuming.

Expected Outputs:

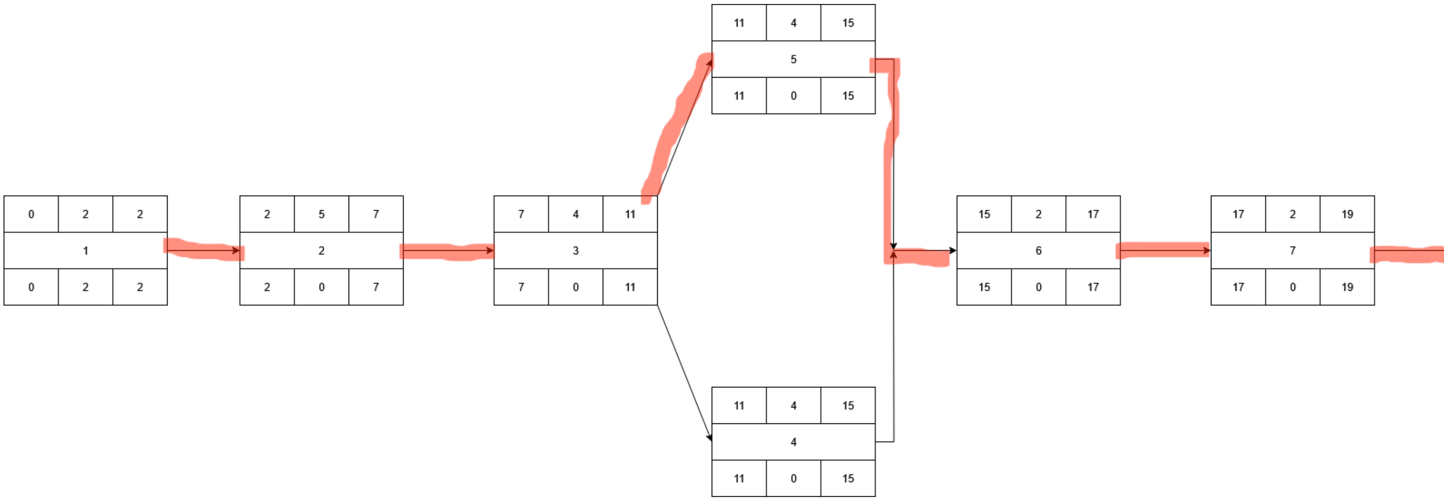
1. Game should pause and player/enemy position should not change.
2. Game should resume and player/enemies should move freely.
3. Game should return to main menu screen and reinitiate gameplay

5. Timeline ________/10

Work items

Task	Duration (Hours)	Predecessor Task(s)
1. Requirements Collection	2	-
2. UI Design	5	1
3. Script Devlopment (buttons)	4	2
4. Integration with Game Logic	4	3
5. Settings Implementation	4	3
6. Automated Testing	2	4,5
7. Final Integrations	2	6
8. Release of Game to the Public	1	7

Pert diagram



Gantt Timeline

Hours (right) Job (below)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	■	■	■																		
2				1	■	■	■	■													
3									2	■	■	■									
4													3	■	■	■					
5													3	■	■	■					
6																	4/5	■			
7																			6	■	
8																					7