Milestone 2: Test Plan

Mark Graven

Southern New Hampshire University

**Contents:**

**RTM / Program Logic --------------------------------------------------------------- [**[**Page 3**](#RTM)**]**

**Bug Testing / Reporting --------------------------------------------------------------- [**[**Page 5**](#Bugs)**]**

**Runtime Error Handling --------------------------------------------------------------- [**[**Page 5**](#Bugs)**]**

**Control Setup/Interface --------------------------------------------------------------- [**[**Page 6**](#Controls)**]**

**Image References --------------------------------------------------------------- [**[**Page 7**](#Images)**]**

**RTM: Contains Program Logic Requirements from GDD**

****

**Program Logic Testing:**

* **Field Themed (ID1) and level is Fenced (ID2)** will be verified by test case (TC) 1 and 2 that are set to explore the playfield and verify the fence is not able to be jumped/passed to maintain the integrity of the playfield. (Image 1)
* **Boulders/Trees (ID3)** will be verified by TC 2 and TC 7 to the extent that the aesthetics do not inhibit ability to play the level and that they add to the experience while working on the quest. (Image 1)
* **Collectible Fuzzies (ID4) and SFX (ID6)** will be verified by TC 4, 5, 7, and 8 by ensuring that collectibles are only collectible (with sound cue) once the round has begun, and that once the level has begun they are able to collect to satisfy the objectives presented to them on the HUD. (Image 2)
* **Random Roaming Fuzzy Spawn (ID5)** will also be verified by TC 4, 5, 7, and 8 to ensure that they indeed randomly spawn and roam (verified by using line tracing from actor creation/start point), and that they’re still able to collect and satisfy the objectives presented on the HUD. (Image 2)
* **Top-Down, First-Person, or Third-Person Interface (ID7)** will be verified by TC1 during exploration of game environment. (Image 3)
* **Mouse, Keyboard, or Combination of Controls (ID8)** will be checked and verified by TC 1, 3, 7, and 8 by playing the game. Refer to the “*Control Setup / Interface*” section (Pg. 4) for specific gameplay controls. (Image 4 and 5)
* **Random Objectives and interface that shows objectives (ID: 9)** will be checked by TC 7 and 8 by initiating/completing multiple tests of the quest. (Image 6, 7)
* **Count Down Timer (ID: 10)** will be checked by TC 6, 7, and 8 to ensure enough time (but not too much) is available to complete the quests, and that the timer stops on quest completion. (Image 3)
* **Suitable for Casual Gamers 5+ (ID: 11)** will be checked by TC 1, 2, 6, 7, 8 by playing through objective and ensuring that the level is inviting, and objectives aren’t outrageous/too hard. (Image 1, 2, 3)
* **Ambient noise/music included (ID: 12)** Will be verified by TC 1 and 7 as the music is designed to change once the quest objectives are obtained. Users will be required to acquire the quest objectives to listen for audio cue changes.
* **Successful or Unsuccessful ratings (ID: 13)** will be verified by TC 6 and 8 to ensure both are implemented. TC6 will allow the timer to expire without completing objectives, whereas TC 8 will be required to complete objectives in the given time. (Image 8, 9)
* **Successful rating that rewards a Fuzzy (ID: 14)** will be verified by TC 8 by completing the assigned objectives within the time limit. This test will verify the user must complete all objectives within the time limit in order to receive a successful rating. (Image 8)
* **Unsuccessful rating only occurs when timer expires (ID: 15)** will be verified by TC 6 which will let the timer expire. To reduce testing time, custom time-limits will be introduced for testing purposes only.

***Bug Testing Plan***

The **Bug Testing** requirements for each test case are as follows:

* Each ID will be tested by each test case (TC) no less than **(10) ten** times per assigned test case.
* The ID will perform as indicated with **no deviation** to pass the test case scenario.
  + If **a deviation from indicated results occurs** it shall be recorded in detail, including all steps to recreate said deviation and the local copy of the RTM will be updated with a “F” and submitted with the suspect bug to source control for review.
    - Once the suspected bug has been identified and corrected, the RTM will be updated as needed to reflect any potential impact to other ID’s and **all impacted TC’s will be evaluated again to ensure integrity.**
      * If needed **additional ID’s** and or TC’s may be created.
  + **Else with no deviations occurring** within the assigned test cases the RTM shall be locally updated to reflect a passing mark of “P” and submitted to source control for review.
* Once all ID’s have passed and the master RTM reflects all passing marks the matrix will be retained for additional scenarios as the client requests.

***Runtime Error Testing***

For **Runtime Errors** I will utilize a combination of:

* Error logs following PIE to isolate which BPs are causing the problem.
* Real-time monitoring of BPs as events execute
  + If the issue isn’t easily visible a breakpoint will be added to step through the event.
    - Once updated, re-evaluation will occur in a PIE session.

***Control Setup / Interface***

**Directional controls** will be set up using the industry standard W-A-S-D format for directional movement control to maximize initial user comprehension.

**W – Forward || A – Turn/Strafe Left || S – Backwards || D – Turn/Strafe Right**

**Camera/Look Control** will be handled utilizing a mouse for the purposes of the demo.

**Interaction Controls** will be implemented so the user can engage with objective activation points (signposts). The **‘E’** key will be the interact key as it is immediately accessible alongside the *directional controls*and doesn’t require the user shift their focus off the screen to interact.

**Jumping** will be controlled by the spacebar and will not be a required interaction to complete the objectives. Its inclusion is to increase the feeling of free movement and ultimately increase the feel of the game for players.

**Menu navigation** will be controlled utilizing the mouse while the menus are on screen. When in a menu the mouse will unlock/appear and allow navigation of the menus.

**Images:**

**Image 1:**

****

**Image 2:**

****

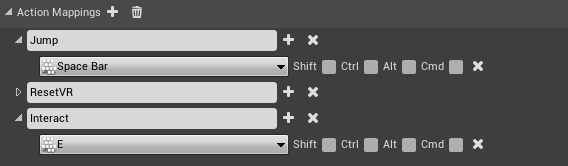
**Image 3:**

****

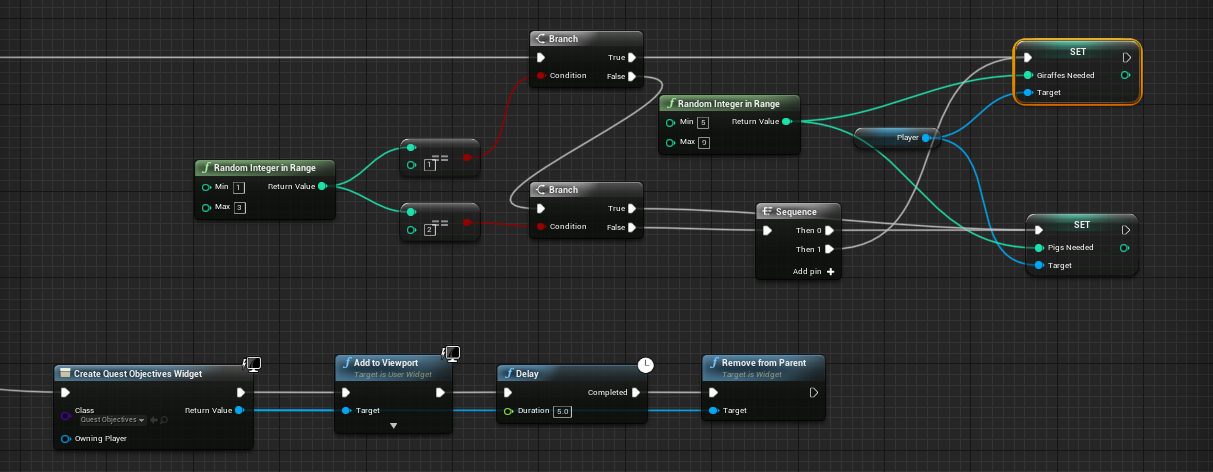
**Image 4:**

****

**Image 5:**

****

**Image 6:**

****

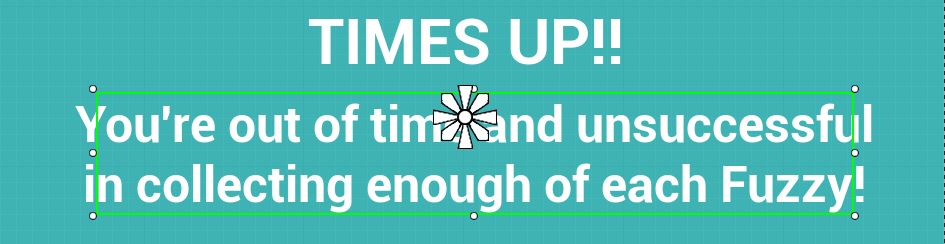
**Image 7:**

****

**Image 8:**

****

**Image 9:**

****