Test Plan for Tanks Game Project

The end users can move and attack other tanks using different tank models and weapons.

Functional Testing

Check whether following are working properly:

Create New Account

- Verify that user can create a new Tanks game account.

Tanks Customization

- Verify that the end user can choose which tank model to use for battle.
- Verify that the end user can equip any weapon onto his or her tank.
- Verify that the end user can either be Player One or Player Two with a distinct color.

Tanks Movement

- Verify that the end user can move his or her tank backward or forward using the keyboard commands.
- Verify that the end user sees his or her tank properly on the screen.
- Verify that the end user's tank is highlighted with a halo circle.
- Verify that the end user's tank and shells do not hide behind other objects on the game screen.

Tanks Combat

- Verify that the end user can successfully point and shoot shells at his or her opponent with the direction of the mouse click.
- Verify that the end user can successfully see an explosion animation when his or her tank is losing health.
- Verify that the end user's tank decreases health points when it takes damage.
- Verify that the end user can visibly see shells fired at his or her opponent on the game screen.
- Verify that the end user's tank gets minimum damage for far range attacks and maximum damage for close range attacks.

Tanks Manager

- Verify that the end user's tank spawns at the same location on the game screen.
- Verify that the game manager displays the proper camera angle for each player.
- Verify that the game manager knows about the three stages of round starting, round playing, and round ending.
- Verify that the game manager displays the relevant messages on the screen for the game stage.
- Verify that the game manager calculates the remaining health points and determines the game winner after a round has ended.
- Verify that the game manager plays the proper audio clip for tank combat and movement.
- Verify that the game manager plays the proper audio click for background music.

Bugs that Impact User Experience

- Players cannot move their tanks in the game using the keyboard.
- The tanks are missing on the spawn points.
- The audio does not work for background, tank movement, and tank shooting.
- Players cannot customize their tank with different weapons.
- Players cannot customize their tank with different models.
- Tanks' health is never reduced to 0 after significant shell damage.

Resources

Tanks

https://www.assetstore.unity3d.com/en/?_ga=1.55059691.1122432568.1485215392#!/content/46209/

https://unity3d.com/learn/tutorials/projects/tanks-tutorial

Compatibility Testing

The Tanks game is compatible with different browsers such as MS Internet Explorer, MS Edge, Mozilla Firefox, Google Chrome, and Apple Safari. Check with stakeholders if any new browser

versions need to be supported or if any older browser versions can be removed from the browser versions matrix.

Device Testing

The Tanks game is compatible on iOS (iPad/iPad Mini), Android (Samsung, Asus), Windows, and Amazon Fire tablets. I would pick a few common tablets based on different OS with different screen sizes to cover the range of how it renders.

The Tanks game is compatible on iOS, Android, and Windows phones. I recommend 7 and 7+ for iOS client. I would use Google Pixel, Samsung Galaxy for Android client. I would use the latest Windows Lumina model for testing.

I can consult with the Project Manager or Operations to see if we have statics of what devices are being used the most for the Tanks program.

Conformance Testing

The Tanks game design meets guidelines, themes, and style for Unity games.

Load Testing

The core functionality should handle the same load for new and existing users on Tanks webpage. Check with product owner and stakeholders what the desired concurrent number of users should be and what the response time for an user should be. Test that the game behaves at these ranges per a predefined tolerance.

Localization Testing

- Verify that the text is translated in all supported languages for the Tanks game home. Check with product owner and stakeholders if there are any new languages that need to be supported

Accessibility Testing

- Verify that the text on all of Tanks game environment is easy to read aka: can be read by different screen sizes like desktop browsers, tablets, and phones.

Security Testing

Perform tests and verify that user accounts' password and credit card information is properly encrypted.

- The tester queries the database and ensures that the data is in an encrypted form in the database.
- For example, the end user pays for custom tank models or weapons. The end user should verify that when the end user submits the information on the form, these parameters are not displayed in the URL.

Brute Force Attacks

- Verify that users will experience temporary account suspension for invalid login attempts.

SQL Injection and Cross Site Scripting

- Verify that the maximum length for input fields are properly defined and implemented by developers.
- Verify that this website discards script redirects from unknown and untrusted applications For example, QA can use Firebug tool to inspect and debug requests on the Tanks game webpage.

Regression Testing

It is not applicable if this is a new feature. For an existing feature, make sure the new functionality does not break existing functionality.

For example, if the feature was improving the shell fire speed. QA should have a set of test cases related to shell fire. He or she can compare the original and new shell fire speed. QA can check that there are no regressions where the actual set of results differs from the expected set

of results. Any deviations should be flagged for the engineering team to decide if this is now expected behavior due to the new design or a true regression.

System Testing

Depending on the feature and changes it could be covered by functionality testing. If the feature introduces a new system component (like supporting a new browser) then the test plan will need to be modified to handle the new system component.

Unit Testing

Unit tests will be written by developers for the sections under Functional Testing if applicable.

Automation Testing

QA or Engineers in Test can write automation tests for tank movement, tank health, and tank combat using mock tanks.

- Mock tanks will make sure that two tanks on the screen can move forward and backward on the game screen.
- A mock tank can remain stationary and inflict damage thus decreasing the opposing tank's health.
- A mock tank can move onto the screen and inflict damage thus decreasing the opposing tank's health.

These examples of automated test cases can be used for smoke testing too.

Project Allocation

3 UI Designers for both iOS/Android clients and webpages.
3 UI Developers for iOS client
3 UI Developers for Android client
2 QA Automated Engineers for mobile apps and web browser.
2 QA Manual Engineers for mobile apps and web browser.
2 Backend Developers

1 DevOps Engineer
1 Tech Writer
1 QA Lead
1 Architect
1 Product Manager/Scrum Manager

Sprint Cycle

2 week sprints includes all code reviewed and checked end before code freeze (last sprint day)
Android candidate released immediately to Play Store after two weeks.
iOS candidate will release to iTunes Store after one week of Apple app review is completed.
Backend candidate released daily for small features and two weeks for large features.

Meetings

Daily Scrum
Product Planning Meeting
Development Planning Meeting
Feature/Backlog Refinement Meeting
Sprint Review Meeting
Retrospective Meeting