

RoboTanks

Test Plan & Specification

Intro

RoboTanks is a web-based game aimed towards Freshmen in Computer Science. The user programs a tank with strategic AI and battles against other tanks. At this point, we have planned the features of the game, scheduled when they will be completed, and designed how they will be implemented. This document treats the testing of the features. It specifies how we will test that each feature has been correctly implemented.

Current Testing

Before getting into testing specific features, we have already been doing small-scale testing along the way. We use Git for version control and Trello for task assignment and bug tracking. Our development team is divided into backend and frontend teams. A developer codes in their own branch, branched off of the backend or frontend branch. When the developer finishes their task, they do basic manual unit testing and request that one of their teammates review their code. Once the code review has passed, the developer merges their branch into the backend or frontend branch. The backend and frontend leaders are responsible for merging their branches into a dev branch, resolving conflicts and performing basic manual integration tests. The project leader is responsible to merge the dev branch into the master branch, which is what is actually run on a public server.

Specific Feature Tests

These tests will verify correct functionality of each feature. The specific test is the steps to perform, and the expected outcome is the required behavior to pass the test.

1.1.1 Create a new user, setting the username and password

Test

- At the home page, select register
- Enter a username and password

Expected Outcome

- You should be taken to the Your Tanks page

1.1.2 Check if a username has already been taken

Test

- Register a username and password that you have already registered

Expected Outcome

- An error should pop up "Username already taken"
- You should remain on the register page

1.2.1 Login using their username and password

Test

- If you are logged in, log out
- Log in with your username and password

Expected Outcome

- You should be taken to the Your Tanks page

1.2.2 See a message if they have the wrong username or password

Test

- At the home screen, enter a username and password that are not registered

Expected Outcome

- An error should pop up "Wrong username or password"
- You should remain on the home page

2.1.1 Create a new tank and give it a name

Test

- Select Upload a Tank
- Enter a name for the tank
- Leave the example tank in the coding window
- Select Add Tank

Expected Outcome

- You should be taken to the Your Tanks page
- The tank you just added should be shown

2.1.2 Choose the tank's class out of given options

Test

- Select Upload a Tank
- Enter a name for the tank
- Modify the example tank to extend HeavyTank instead of Tank or BasicTank
- Select Add Tank

Expected Outcome

- You should be taken to the Your Tanks page
- The tank you just added should be shown

2.1.3 Upload code for that tank

This is tested in 2.1.2

2.1.4 Check the uploaded code to see if it will compile

Test

- Perform the steps in 2.1.2

Expected Outcome

- You should not see an error that the tank failed to compile

2.2.1 Download the code for one of their tanks in it's original form as they uploaded it

Test

- Select Your Tanks
- Select a tank that you have added
- Select Download Tank

Expected Outcome

- The tank's code should be saved to your machine

2.2.2 Change the tank's class

Test

- Select Your Tanks
- Select a tank that you have added
- Modify the tank's code to extend HeavyTank instead of Tank or BasicTank
- Select Save Tank

Expected Outcome

- The tank's code should now display your changes

2.2.3 Replace a tank's code with a new upload

This is tested in 2.2.2

2.2.4 Check the uploaded code to see if it will compile

Test

- Perform the steps in 2.2.2

Expected Outcome

- You should not see an error that the tank failed to compile

2.3.1 Delete a tank

Test

- Select Your Tanks
- Select a tank that you have added
- Select Delete Tank

Expected Outcome

- The tank should no longer be shown in your list of tanks

2.4.1 See a list of all their tanks

Test

- Select Your Tanks

Expected Outcome

- You should be taken to the Your Tanks page
- The tanks you have added should be shown

2.4.2 Rename a tank

Test

- Select Your Tanks
- Select a tank that you have added
- Enter a different name in the tank's name field
- Select Save Tank

Expected Outcome

- The tank's new name should now be shown in place of its old one

2.4.3 Look at the source code for a tank (without downloading it)

Test

- Select Your Tanks
- Select a tank that you have added

Expected Outcome

- You should see the tank's source code

3.1.1 Create a new game and give it a name

Test

- Select Join a Fight
- Enter a name for your game

Expected Outcome

- The fight you just created should be shown in the game list

3.2.1 See a list of games with open spots

Test

- Select Join a Fight

Expected Outcome

- You should be taken to the Start a Game page

3.2.2 Join an open game with a selected tank

Test

- Create a game
- Select Join
- Select your tank
- Repeat to join your tank to your game 4 times

Expected Outcome

- The number of open spots should decrease by 1 each time
- When you take the last spot, the game should disappear from the game list

3.3.1 See a list of games they've joined

Test

- Select See Your Games

Expected Outcome

- You should be taken to the Your Games page
- The games you have joined should be shown

3.3.2 Watch a game

This feature will be tested by the results of tests 3.3.3 – 3.3.8

3.3.3 Watch the movements of each tank in the game over time (animated)

Test

- Select See Your Games
- Select Watch Game

Expected Outcome

- The tanks should visibly move around the board

3.3.4 Watch the movements of bullets over time (animated)

Test

-Select See Your Games

-Select Watch Game

Expected Outcome

-The tanks should visibly shoot bullets at each other

3.3.5 Watch the health of each tank throughout the game

Test

-Select See Your Games

-Select Watch Game

Expected Outcome

-The tanks' healths should visibly decrease when hit by bullets

3.3.6 Change the playback speed of a game ("slow motion" vs. "fast forward")

Test

-Select See Your Games

-Select Watch Game

-Select "Slow" or "Fast"

Expected Outcome

-The movement of the game should visibly increase or decrease

3.3.7 Discern between their tank and enemy tanks (color coordination)

Test

-Select See Your Games

-Select Watch Game

Expected Outcome

-Tanks should be shown with the skin they selected on creation

3.3.8 See any exceptions/errors generated by their tanks during the game

Test

-Join a game with a tank that doesn't have compilation errors but does have a runtime error

Expected Outcome

-You should see an error "Runtime error"

4.1.1 Rotate a full 360 degrees

Test

-Join a game with a tank that turns right every turn

-Watch the game

Expected Outcome

-The tank should turn 90 degrees each turn

-After 4 turns, the tank should have completed 360 degrees

4.1.2 Rotate in increments as small as 90 degrees

This is tested in 4.1.1

4.1.3 Rotate to one of the 4 absolute directions (N, E, S, W)

Test

- Join a game with a tank that turns right until it is facing North
- Watch the game

Expected Outcome

- The tank should rotate until it is facing North, then stop turning

4.1.4 Rotate relative to current direction (Clockwise by 90, Counterclockwise by 180, etc.)

This is tested in 4.1.1

4.2.1 Move forwards or backwards

Test

- Join a game with a tank that moves forward then backward every other turn
- Watch the game

Expected Outcome

- The tank should move both forward and backward

4.2.2 Move a certain number of grid spaces ("move forward 4")

Test

- Join a game with a tank that moves forward 4 spaces then stops (4 moves)
- Watch the game

Expected Outcome

- The tank should move forward 4 turns, then wait

4.3.1 Shoot straight forward in the direction it's facing

Test

- Join a game with a tank that shoots
- Watch the game

Expected Outcome

- The tank should shoot straight forward

4.4.1 Wait until their next turn

Test

- Join a game with a tank that waits
- Watch the game

Expected Outcome

- The tank should stay still, while the other tanks move

4.5.1 Use action points by choosing which action to do

Test

- Join a game with a tank that moves forward/backward and a tank that turns left/right
- Watch the game

Expected Outcome

- The turning tank should turn twice as frequently as the moving tank moves
- This is because it takes half as much AP to turn as it does to move

5.1.1 See a list of possible tank moves

Test

-Select Manual

Expected Outcome

-You should see a section on possible tank moves

5.1.2 See how many action points each move will cost for each tank class

Test

-Select Manual

Expected Outcome

-You should see a section on action points for each tank class

5.1.3 See examples of tanks that work

Test

-Select Manual

Expected Outcome

-You should see a section on example tanks

5.2.1 Look up the rules for the game

Test

-Select Manual

Expected Outcome

-You should see a section on game rules

5.2.2 Look up rules for how action points are used

Test

-Select Manual

Expected Outcome

-You should see a section on action points

6.1.1 See the number of games they've won

Test

-Select Armory

Expected Outcome

-You should see a statistic for how many total games you have won

6.1.2 See the number of games they've lost

Test

-Select Armory

Expected Outcome

-You should see a statistic for how many total games you have lost

6.1.3 Total tanks killed

Test

-Select Armory

Expected Outcome

-You should see a statistic for how many total tanks you have killed

6.1.4 Total tanks lost

Test

-Select Armory

Expected Outcome

-You should see a statistic for how many total tanks you have lost

6.2.1 See the number of games a tank has won

Test

-Select Armory

-Select a tank

Expected Outcome

-You should see a statistic for how many games the tank has won

6.2.2 See the number of games a tank has lost

Test

-Select Armory

-Select a tank

Expected Outcome

-You should see a statistic for how many games the tank has lost

6.2.3 See the number of other tanks a tank has killed

Test

-Select Armory

-Select a tank

Expected Outcome

-You should see a statistic for how many tanks the tank has killed

6.2.4 See the number of times a tank has been killed

Test

-Select Armory

-Select a tank

Expected Outcome

-You should see a statistic for how many times the tank has been killed

7.1.1 (NOT) Create any additional files on the server

Testing method to be determined

7.1.2 (NOT) Delete anything on the filesystem on the server

Testing method to be determined

7.1.3 (NOT) Get access to a shell on the server

Testing method to be determined

7.2.1 (NOT) See the code for other users' tanks

Testing method to be determined

7.2.2 (NOT) Submit code for other users' tanks

Testing method to be determined

Alpha Testing

Once individual feature testing is complete, we will move on to alpha testing the product as a whole. The game is already on a public server, so we have been experimenting with it as we go, but we will shift our focus to testing all of the features together as a user would.

Beta Testing

Once we are satisfied with the product's functionality and security, we will share our server's address with people outside of our team. This will include anyone from friends, to classmates, to CS Freshmen. We will have an available line of communication to receive feedback, and will adjust the game until we are comfortable to officially release it.