

FVS Test Plan

1 TEST PLAN IDENTIFIER **TEST4**

2 RELATED DOCUMENTS

- [1] Specification Document (Assessment 1 Deliverable)
 - Requirements
 - Project design
 - Project plan
- [2] Assessment 3 Test Plan
- [3] IEEE. "Test Plan Outline (IEEE 829 FORMAT)". Available:
<http://www.computing.dcu.ie/~davids/courses/CA267/ieee829mtp.pdf>

3 INTRODUCTION

This is the third master test plan for the Trains Across Europe (TaxE) game. This plan will address the features implemented in the third iteration of the game and, where applicable, their effects on those implemented in the first and second iterations.

4 SOFTWARE RISK ISSUES

We have relied on libraries from both the Gradle and LibGDX frameworks to ease the implementation of many of the graphical aspects of the game. We are therefore unable to test details within these external libraries.

5 NEW FEATURES TO BE TESTED

The following is a list of the new features to be focused on during this phase of testing, as per the specified requirements changes:

1. Replay mode

- As per user requirement 8; which requires the addition of a replay mode to the game, such that a selection set of turns can be recorded and then ‘replayed’ quickly.

2. Track modification

- As per user requirement 9; which requires to provision of a mechanism to allow players to dynamically add new track, and remove existing track from the game.

3. General enhancements

6 APPROACH

We will make use of a mixture of testing techniques in order to test comprehensively:

- **Unit testing** will be performed to test specific code where possible; we will make use of the JUnit testing framework supported by both the Eclipse and IntelliJ IDEs. This will be done by the developer(s) responsible for each part of the code being tested; Manfred for track modification, James for replay and Owen for general enhancements.
- **System testing** will be performed to evaluate the system’s compliance with its specified *requirements*, this will be achieved by running the application and performing tests manually, in black-box manner. This will be done by our system testing team, Tyler and Ellis for this assessment - once the relevant features have been implemented.

- After the respective developer has done low level testing of their feature, they will then test their entire feature within the context of the game as **integration testing**. For example in the case of track modification; the multistep process will include: adding a connection resource, the back-end registering that a new connection exists and the resource being removed from the player that used it.
- Then we will ensure that our newly implemented features do not break existing parts of the project, through **regression testing**. To include the existing unit tests, and system testing when a feature hasn't been clearly demonstrated as still working elsewhere.
- Lastly we will ensure that our product works correctly in our ideal scenario, through **acceptance testing**. By playing through a few iterations of the game on a lab machine, using as many features as possible, and recording how fit for purpose the product is.

7 ITEM PASS CRITERIA

- The unit testing will be complete once all of the JUnit tests have passed, and we can therefore expect a working implementation.
- We will class system testing as complete once we are confident that all of the requirements that we set out to test in this iteration are fulfilled consistently.

8 SCHEDULE

The timescale for this iteration of the project is 10 weeks and the testing must also be completed within this timescale. Lower level testing such as unit and integration testing will be performed and documented alongside implementation where possible, and higher level testing such as regression and acceptance testing will be performed after.

EVIDENCE OF TESTING

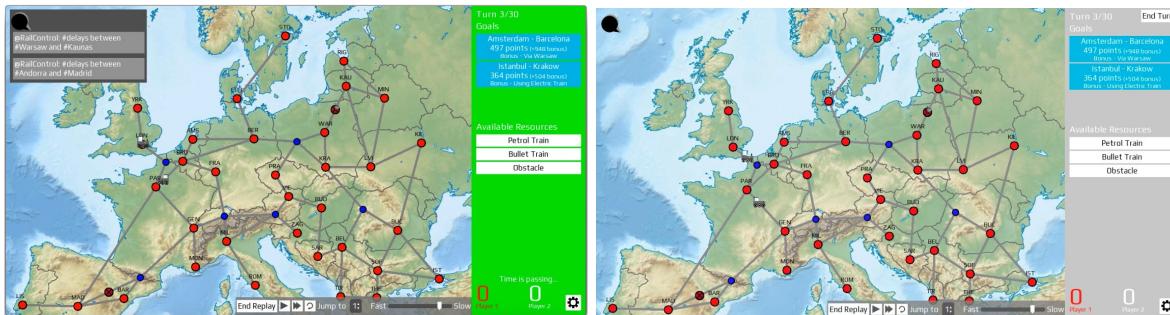
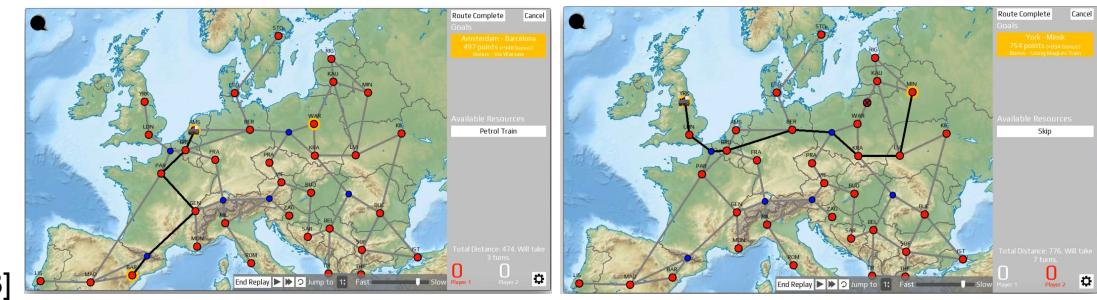
1 UNIT TESTING

Test name	Test Description	Result
defaultVolumeTest	Tests that when a SoundController is initialised with no settings file that the default volumes are used.	Pass
saveSettingsAndLoadTest	<p>Modify the values of sound a music volume before writing out to the settings file. The SoundController is then re-initialised and volumes confirmed to be correct.</p> <p>The implementation of this feature was not originally known so the test could only be added after the feature had been implemented.</p> <p>Test code is available in Appendix A.</p>	Pass
addClickToReplayTest	Confirms that the ReplayManager has no clicks stored in it at first and then adds a new click event to it before confirming that it has been added to the list by testing that actions exist that need to be replayed.	Pass
rotationTest	Tests that the rotation algorithm is generating the correct rotation value for the train to use. Sets a train route and then uses the code from the TrainActor to calculate its rotation (noting that angles must be specified counter-clockwise in degrees) and confirm it is correct.	Pass
dijkstraUpdateTest	Tests that implementation of Dijkstras Algorithm is calculating the correct length initially between indirectly connected stations and then that the result given is correctly updated when a new connection is added. If it handles a connection being added, it can be safely assumed it will handle on being removed.	Pass

All Unit tests were conducted during and after the implementation had been completed to ensure that the final deliverable product was functioning correctly.

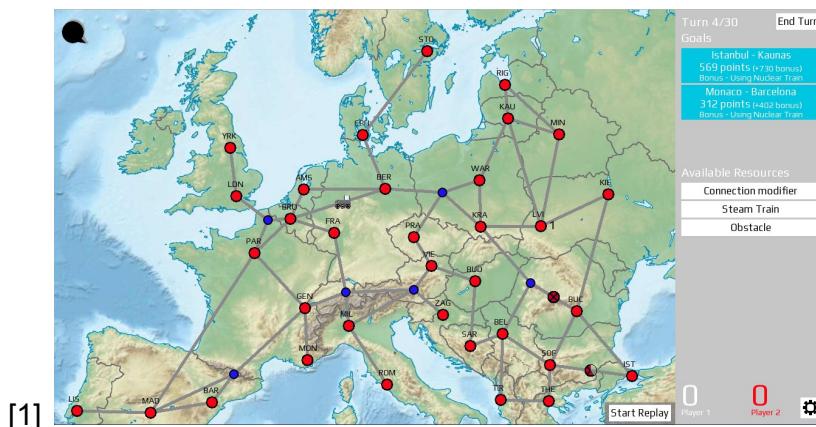
2 SYSTEM AND INTEGRATION TESTING

Test ID	Test Description	Result
1	Replaying a set of turns automatically <ul style="list-style-type: none"> Play through some turns and stop at a point to test [1] Select 'start replay' to enter replay mode [2] Press the  button to start automated playback [3] 	Pass
Evidence		
<p>[1]</p>		



Test ID	Test Description	Result
2	Recorded turns replayed faster than real-time <ul style="list-style-type: none"> Play through 3 turns and record time taken [1] Replay turns on fastest and record time taken to replay [2] 	Fail

Evidence

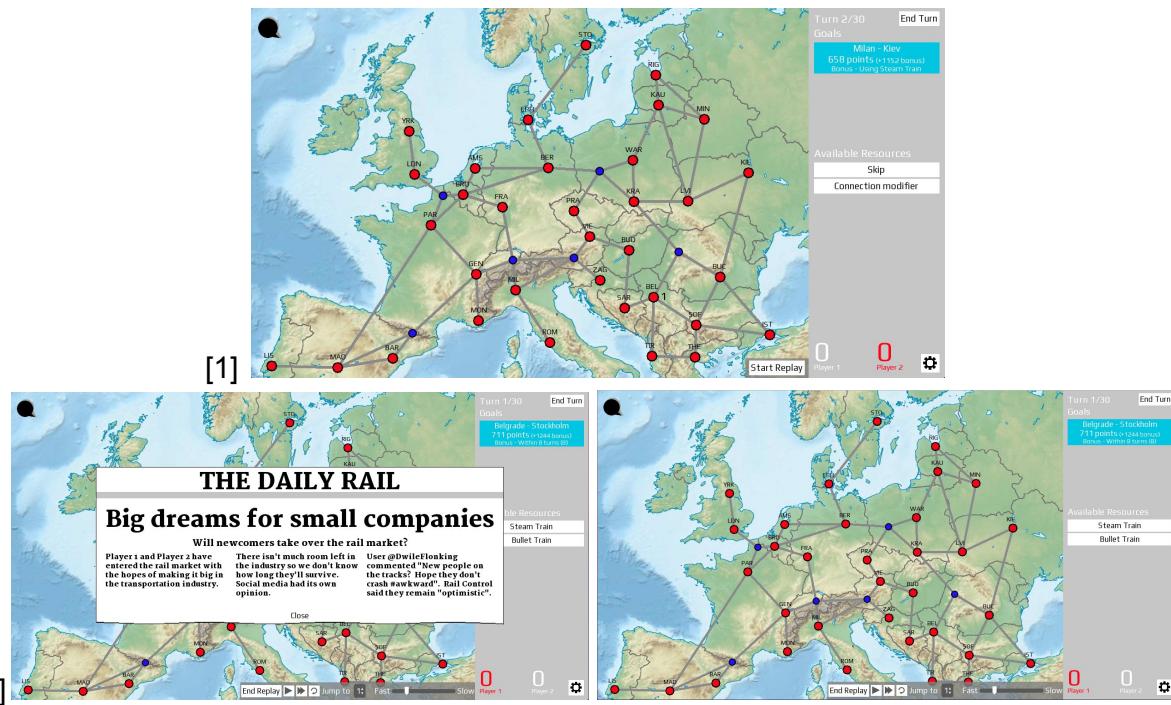


[2] Replay crashed, due to a previously undiscovered bug (Bug 4 in bug report)

Test ID	Test Description	Result
3	Recorded turns replayed faster than real-time (Reattempt) <ul style="list-style-type: none"> Play through 3 turns and record time taken [1] Replay turns on fastest and record time taken to replay [2] 	Pass
Evidence		
[1]	 <p>Turn 4/30 Goals Monaco - Budapest 409 points (via Amsterdam) 1 turn End Turn</p> <p>Available Resources Nuclear Train Petrol Train Skip</p> <p>www.online-stopwatch.com 00:00:42 58</p> <p>Cont... Clear Back</p>	
[2]	 <p>Turn 1/30 Goals Transfär - Stockholm 534 points (via Helsinki) 1 turn End Turn</p> <p>Available Resources Obstacle</p> <p>THE DAILY RAIL Big dreams for small companies Will newcomers take over the rail market? Player 1 and Player 2 have entered the rail market with the hopes of making it big in the transportation industry. There isn't much room left in the industry so we don't know how long they'll survive. Social media had its own opinion. User @TwileEloking commented "New people on the tracks! Hope they don't crash forward". Rail control said they remain "optimistic".</p> <p>Route Complete Hanover - Stockholm 534 points (via Helsinki) 1 turn End Turn</p> <p>Available Resources Obstacle</p> <p>00:00:09 446</p> <p>Cont... Clear Back</p>	

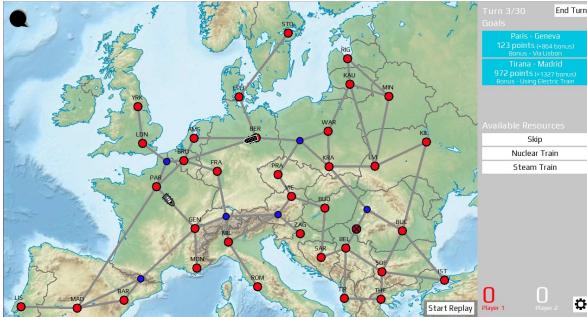
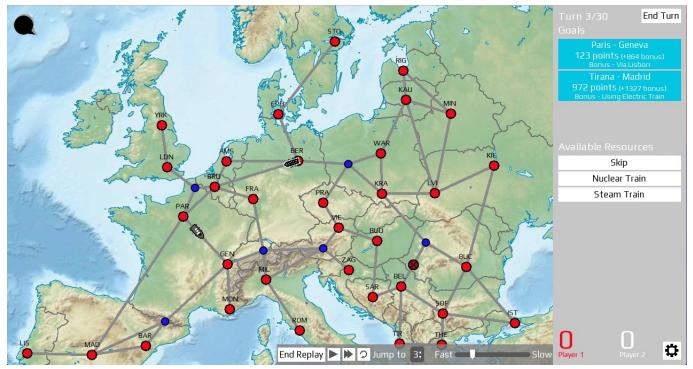
Test ID	Test Description	Result
4	<p>Advance through an individual click in replay</p> <ul style="list-style-type: none"> Play through a turn [1] Enter replay mode and click the button to advance through an individual click [2] 	Pass

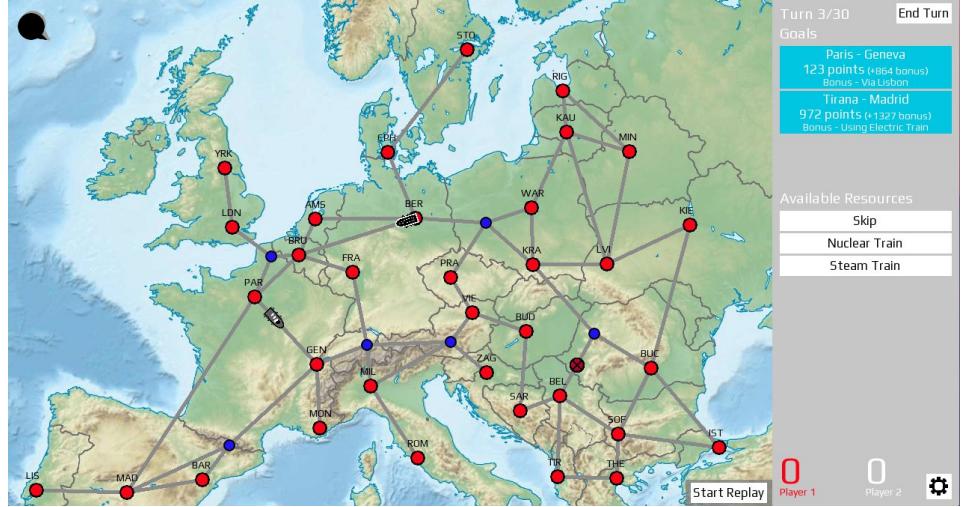
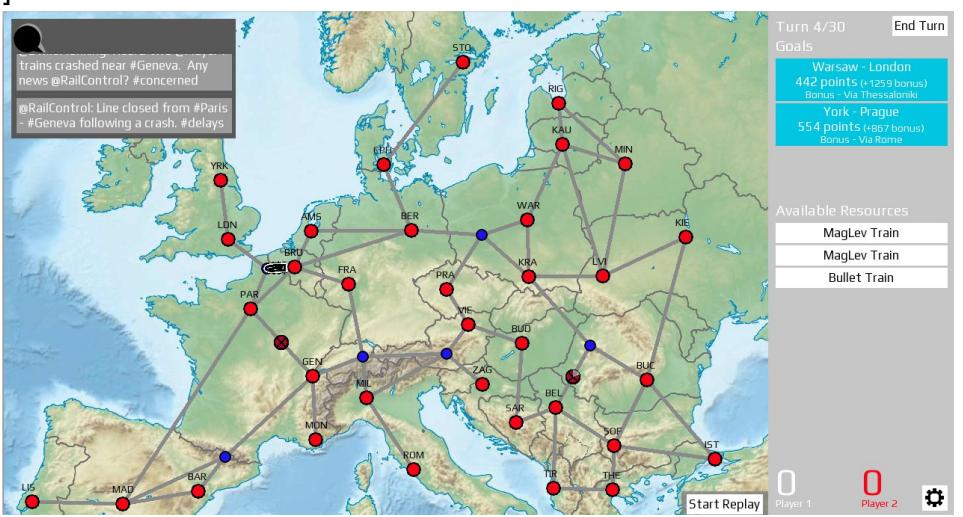
Evidence



Test ID	Test Description	Result
5	<p>Advance through an individual turn in replay</p> <ul style="list-style-type: none"> Play through 2 turns [1] Enter replay mode and click the button to advance through an individual turn [2] 	Pass

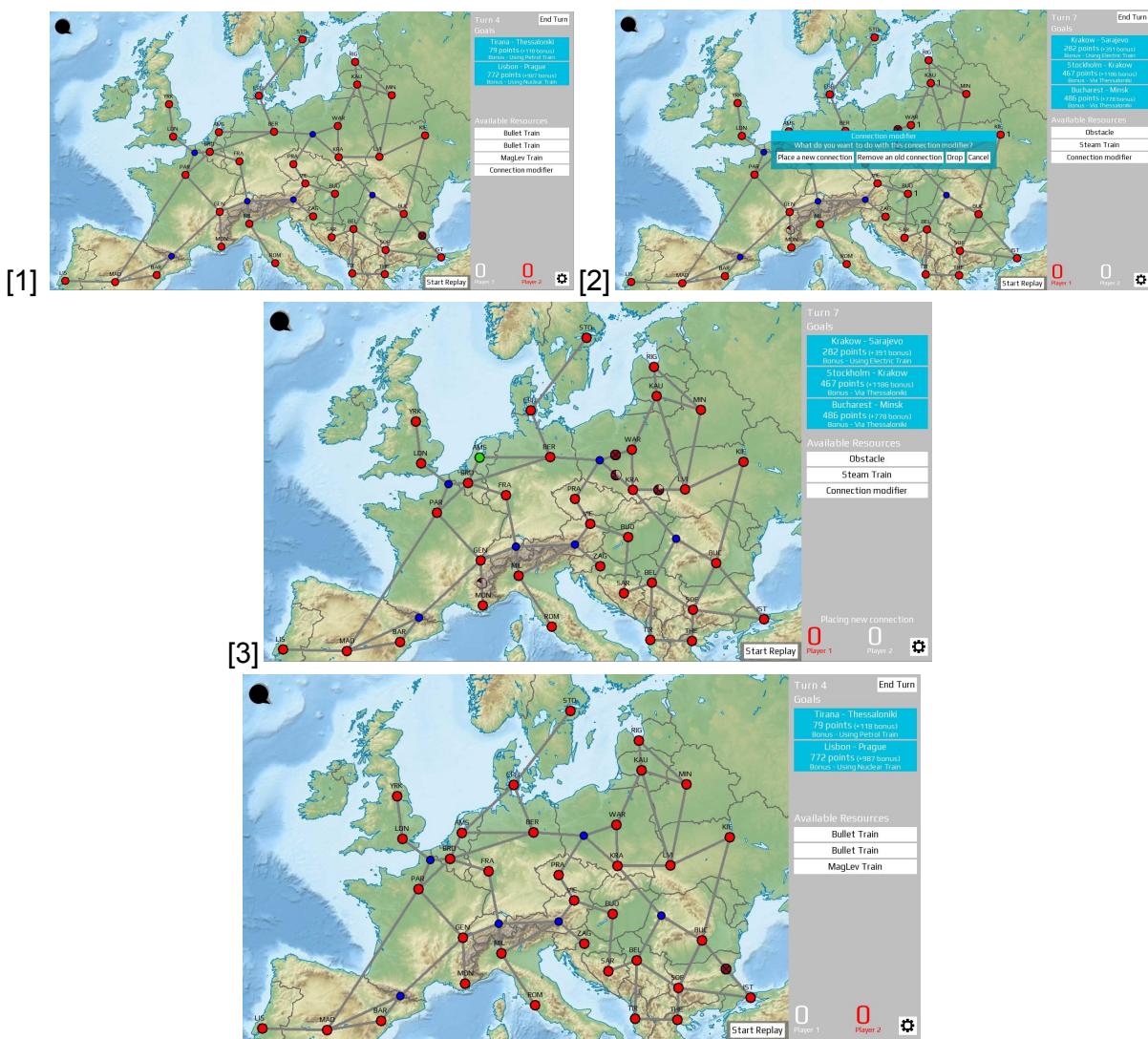
Evidence

	 <p>[1]</p>	
	 <p>[2]</p>	
Test ID	Test Description	Result
6	Jump to a specified turn in the game <ul style="list-style-type: none"> Enter replay mode and use the 'Jump to' control to jump to turn 3 [1] 	Pass
Evidence		
	 <p>[1]</p>	

Test ID	Test Description	Result
7	Exit replay mode and resume normal play <ul style="list-style-type: none"> Enter replay mode [1] Exit replay mode and play a turn [2] 	Pass
Evidence		
	 <p>[1]</p>  <p>[2]</p>  <p>Turn 4/30 End Turn Goals Warsaw - London 442 points (+1259 bonus) Bonus - Via Thessaloniki York - Prague 554 points (+467 bonus) Bonus - Via Rome</p> <p>Available Resources MagLev Train MagLev Train Bullet Train</p>	

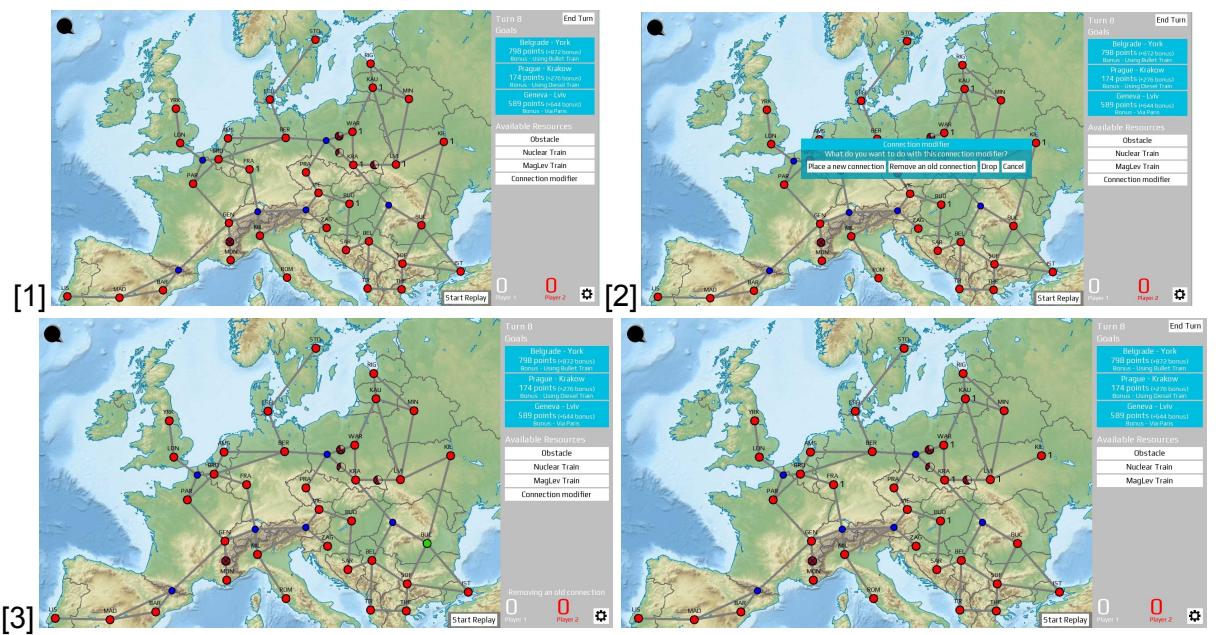
Test ID	Test Description	Result
8	<p>Add a new piece of track</p> <ul style="list-style-type: none"> Click “Connection Modifier” resource [1] Click “Place a new connection” [2] Click on the stations you wish to join [3] 	Pass

Evidence



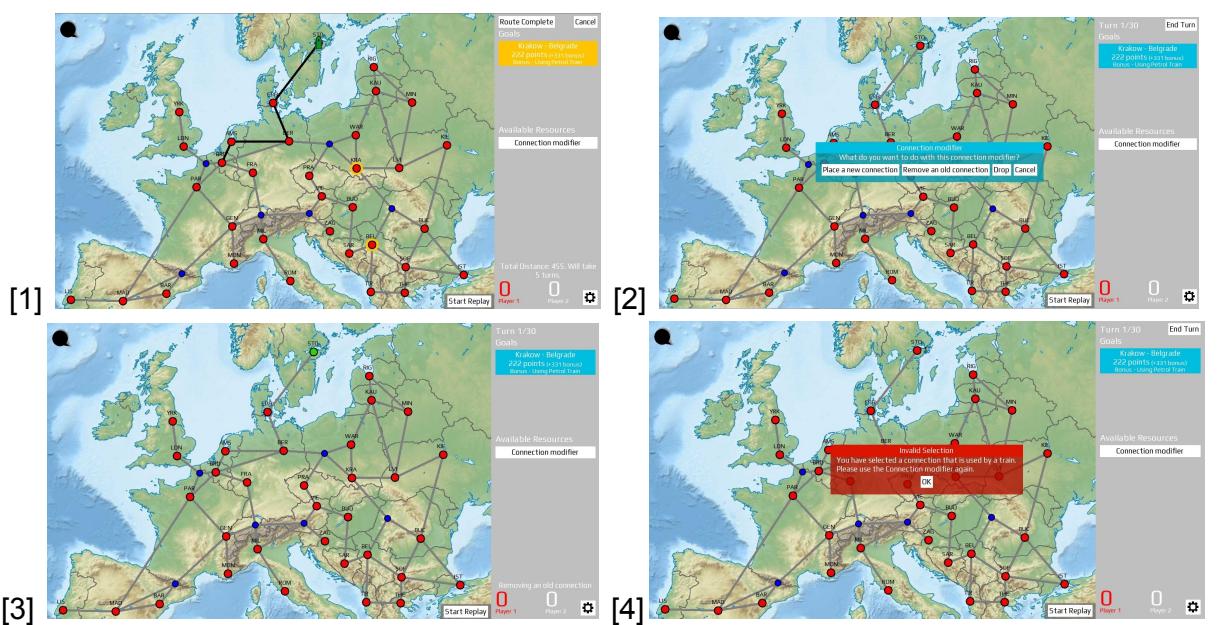
Test ID	Test Description	Result
9	<p>Remove an existing piece of track</p> <ul style="list-style-type: none"> Click “Connection Modifier” resource [1] Click “Remove an old connection” [2] Click on the stations you wish to separate [3] 	Pass

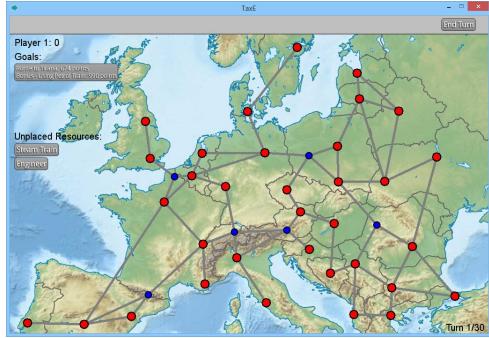
Evidence



Test ID	Test Description	Result
10	Attempt to remove in-use piece of track <ul style="list-style-type: none"> Place and route a train [1] Click “Connection Modifier” resource [2] Click “Remove an old connection” [3] Click on the stations you wish to separate [4] 	Pass

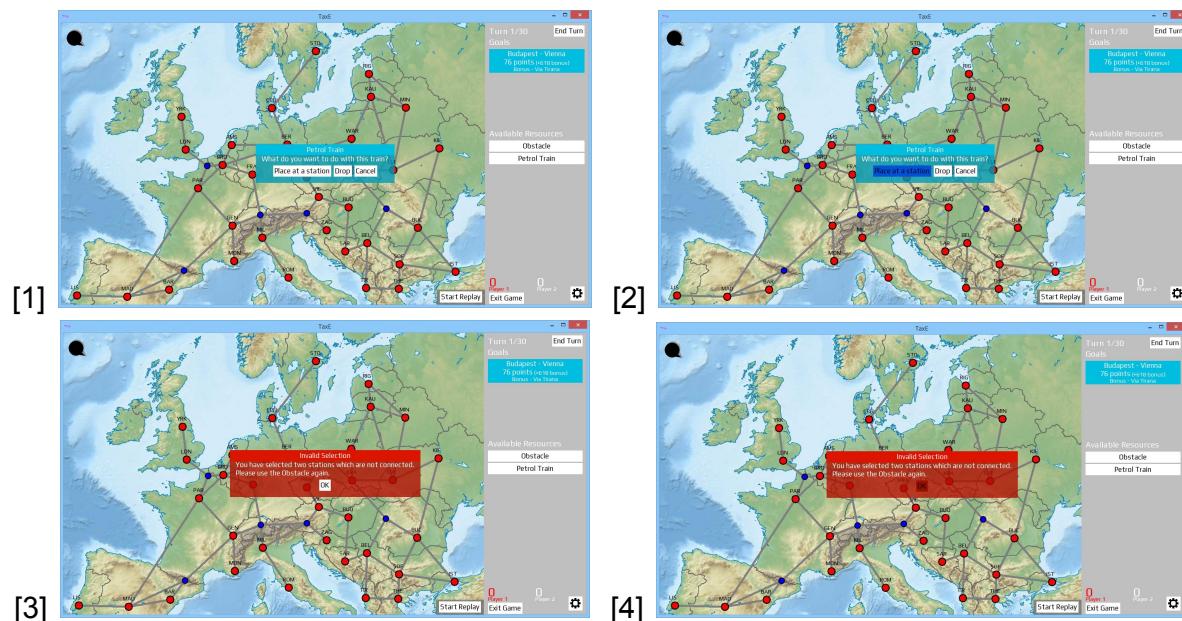
Evidence



Test ID	Test Description	Result
11	Attempt to resize the game window and see scaled content <ul style="list-style-type: none"> Launch the game in default size [1] Resize the window (White bounding lines appear) [2] 	Pass
Evidence		
[1]		
[2]		
Test ID	Test Description	Result
12	Observing differences in UI styles <ul style="list-style-type: none"> Launch previous version (Assessment 3) of game and start game [1] Launch current version of game and start game [2] 	Pass
Evidence		
[1]		
[2]		

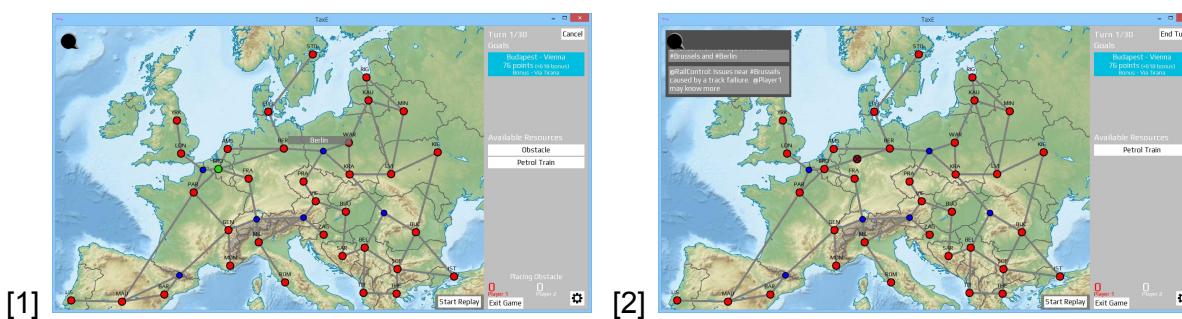
Test ID	Test Description	Result
13	<p>Dialog styling is being carried by Unified Dialog</p> <ul style="list-style-type: none"> Open a TrainResourceDialog box from the resources list [1] Press on button and see background colour changes to match window [2] Repeat with a red dialog and see button changes to red [3 & 4] 	Pass

Evidence



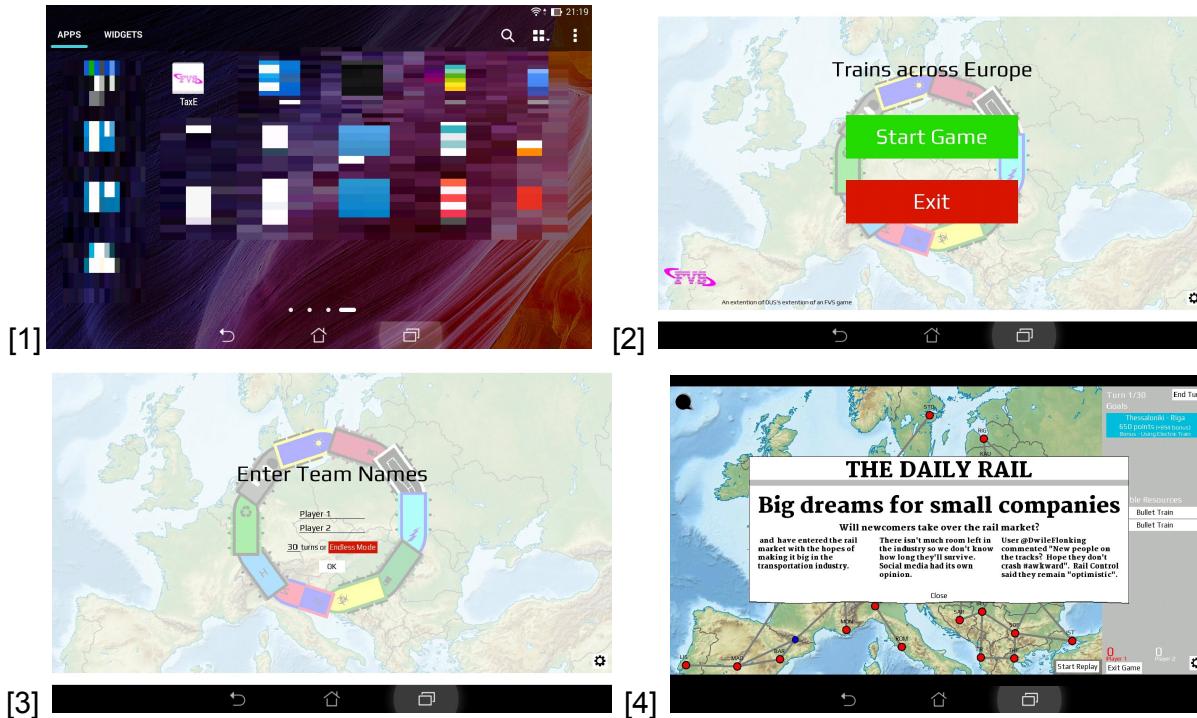
Test ID	Test Description	Result
14	<p>Social media information on obstacle appearance</p> <ul style="list-style-type: none"> Place an obstacle on a connection [1] Message appears in top left hand corner [2] 	Pass

Evidence



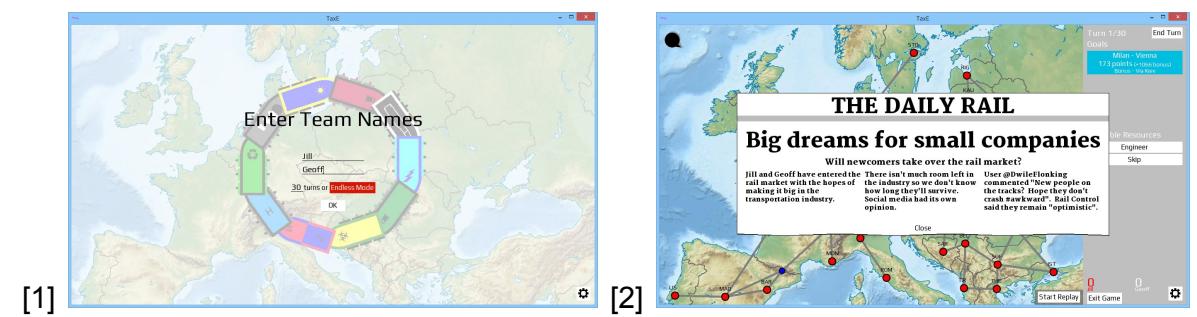
Test ID	Test Description	Result
15	<p>Launch game to GameScreen on Android</p> <ul style="list-style-type: none"> • Launch game from Launcher [1] • Press Start Game [2] • Press OK on SetupScreen [3] • Wait for GameScreen to appear [4] 	Pass

Evidence



Test ID	Test Description	Result
16	<p>Change player names</p> <ul style="list-style-type: none"> • Enter a name for Player 1 and Player 2 on SetupScreen [1] • Press OK and player names are on screen [2] 	Pass

Evidence



3 ACCEPTANCE TESTING

Test ID	Date and time	Tester
1	28/04/2015 - 12:00	Tyler
Results		
<ul style="list-style-type: none">Played through a 20 turn game successfully on a student lab PC, however forgot to test replay before ending the game, which highlighted the inability to replay once the game has ended and thus, replay an entire game.To retest, including the use of the replay feature to ensure all requirements met.As a consequence, a replay button was added to the 'Game over' dialogue.		

Test ID	Date and time	Tester
2	28/04/2015 - 15:15	Ellis
Results		
<ul style="list-style-type: none">Played through a 18 turn game all the way throughWhile replaying turns 6-12 the replay crashed but the error was caughtI was able to finish the game with no issues.		

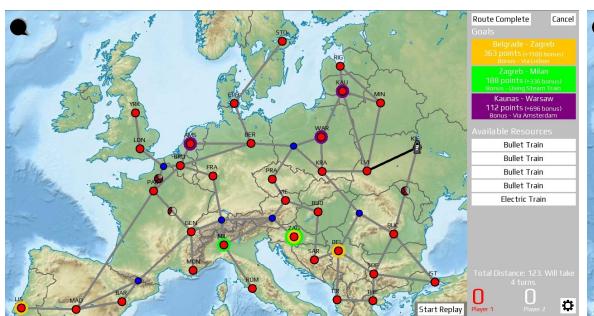
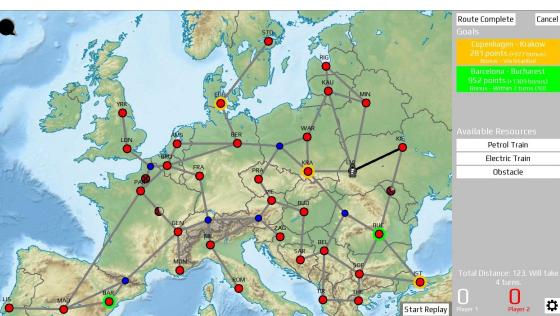
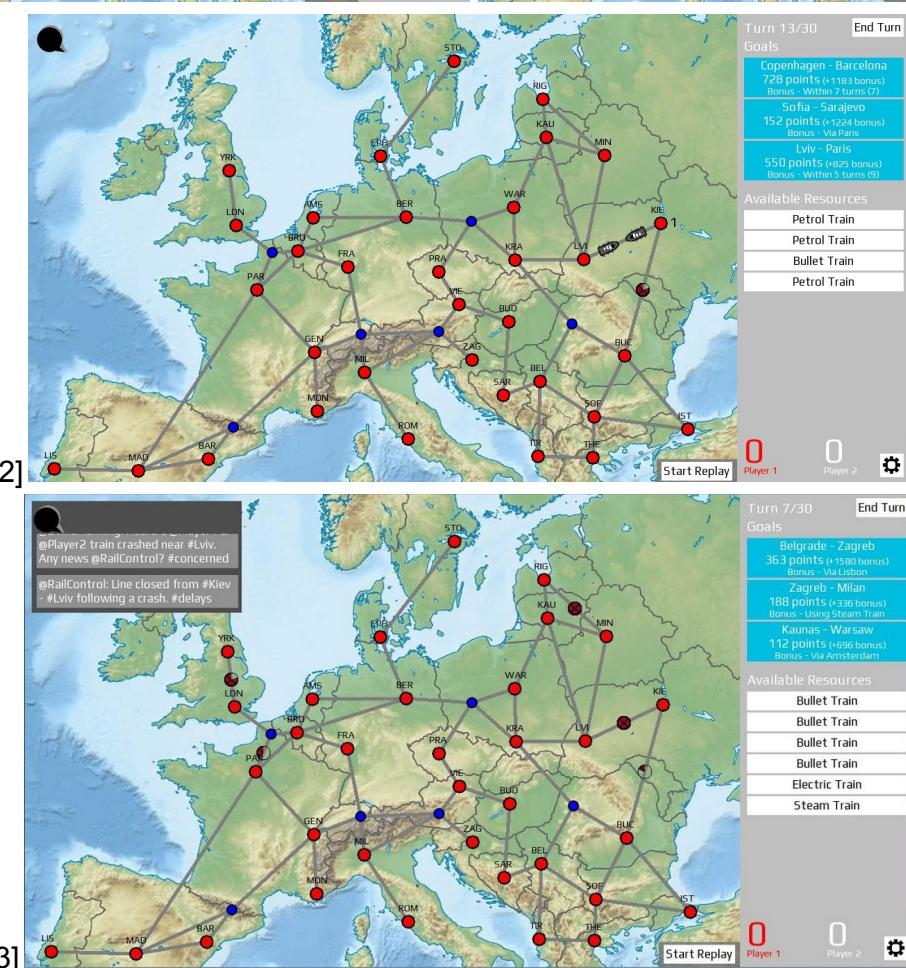
Test ID	Date and time	Tester
3	28/04/2015 - 19:30	Owen
Results		
<ul style="list-style-type: none">Played through a 22 turn game entirely on Android before replaying all turns once the game had ended.The game played and replayed successfully.Replay was entirely linear with no jumping.		

4 REGRESSION TESTING

Previous unit tests were tested to ensure that they passed, if not they were fixed:

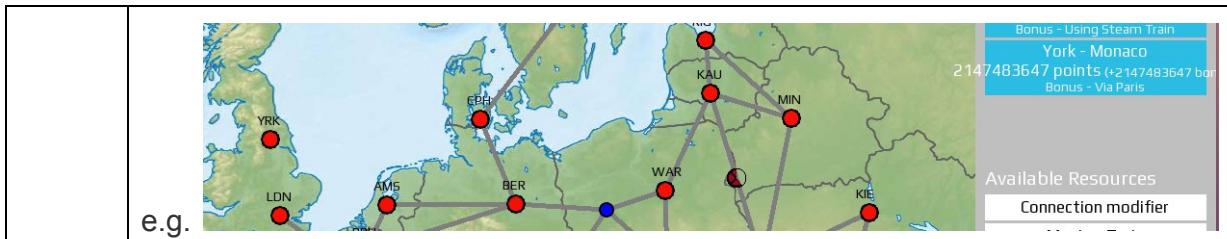
Test Name	Test Class	Description	Result
testInitialisePlayers()	GameTest	Assert that the current player has both goals and resources	Fail
testPlayerChanged()	GameTest	<ul style="list-style-type: none"> Get number of resources and goals of current player. End turn twice to return to initial player Assert that number of resources and goals has increased 	Fail
testInitialisePlayers()	GameTest	These tests failed due to the addition of replay, they were modified to include storing the seed for RandomSingleton, which allowed to tests to function again.	Pass
testPlayerChanged()	GameTest		Pass
canReadFile()	GdxFileTest	<ul style="list-style-type: none"> Pass a json file to the file handler Assert a statement about the length of the file 	Pass
testGenerateRandom()	GoalManagerTest	<ul style="list-style-type: none"> Create a train and two stations Create a goal for moving a train between these stations Create a route for the train between these stations End turn twice Assert that goal has been completed 	Fail
testAddRandomGoalToPlayer()	GoalManagerTest		Fail
testGenerateRandom()	GoalManagerTest	Due to a change in allocating initial goals to players, the assert statements were offset by 1 causing tests to fail. This was rectified to allow tests to pass.	Pass
testAddRandomGoalToPlayer()	GoalManagerTest		Pass
testIsComplete()	GoalTest	<ul style="list-style-type: none"> Checks the completion of a goal before and after the train has reached the destination, knowing it has passed through the start station 	Pass
testCompletedWithinMaxTurns()	GoalTest	<ul style="list-style-type: none"> Simulate the completion of a goal both before and after n turns, where n is the number of turns associated to the bonus 	Pass
testCompletedWithTrain()	GoalTest	<ul style="list-style-type: none"> Create a goal 	Pass

Test Name	Test Class	Description	Result
addStationAndConnectionTest	MapTest	<ul style="list-style-type: none"> Get number of current stations on map Create two new station Assert that number of stations has increased by two Add connection between two new stations Assert that connection exists between the new stations 	Pass
testGetCurrentPlayer()	PlayerManagerTest	<ul style="list-style-type: none"> Get current player then end turn Assert that new player is not equal to the new current player 	Pass
testTurnNumber()	PlayerManagerTest	<ul style="list-style-type: none"> Store the value of the turn number End turn Assert that current turn number is greater than previous turn 	Pass
testGetRandomTrain()	ResourceManagerTest	<ul style="list-style-type: none"> Call function several times and get different results each time 	Fail
testAddRandomResourceToPlayer()	ResourceManagerTest	<ul style="list-style-type: none"> Try to increase the resources past the maximum number of resources Assert that the number of resources is equal to the maximum number of allowed resources 	Fail
testGetRandomTrain()	ResourceManagerTest	These tests were missing a proper setUp() method, upon addition of this the tests passed.	Pass
testAddRandomResourceToPlayer()	ResourceManagerTest		Pass
testDoesConnectionExist()	StationHelperTest	<ul style="list-style-type: none"> Use a connection between two stations that does exist and assert that it does Use a connection between two stations that does not exist and assert that it doesn't 	Pass
StationsTest()	StationTest	<ul style="list-style-type: none"> Create a test station at a specific point on the map Assert that the station location is equal to the specified point Assert that the station name is equal to the name assigned for it 	Pass
finalDestinationTest()	TrainTest	<ul style="list-style-type: none"> Create two stations Add these stations to a route Assert that the route contains two stations Assert that the second station is the final destination in the route 	Pass

Test Description	Result
<p>New trains still collide correctly</p> <ul style="list-style-type: none"> Place and route two trains to collide [1] End turn until the trains collide [2] Both trains get destroyed and an obstacle forms [3] 	Pass
Evidence	
 <p>[1]</p>  <p>[2]</p>  <p>[3]</p>	

5 BUG REPORT

Bug ID	Bug Description
1	<p>Connections between same station</p> <p>The connection modifier allows a player to add a connection between a station and that same station, a check needs to be added to disallow this:</p> <p>e.g.</p> <div style="border: 1px solid #ccc; padding: 5px; background-color: #f9f9f9;"> @Player1: We're building a new @RailControl line from #Prague to #Prague #progress </div>
Resolution	
<p>Extended the check as to whether a connection already exists between the two stations to include whether they are the same station. An invalid selection error is shown and the connection modifier remains available to use:</p>	
2	<p>'Infinite' score for a goal including disconnected city</p> <p>When a city is disconnected from the rest of the map, the score for a goal involving that city is set to max int (2147483647). The city can then be reconnected and that score achieved, making the player pretty much unbeatable. Need to update goal generation to facilitate this:</p>



Resolution

There is now a check to determine whether a station is disconnected, if so, it is not included in any newly generated goals.

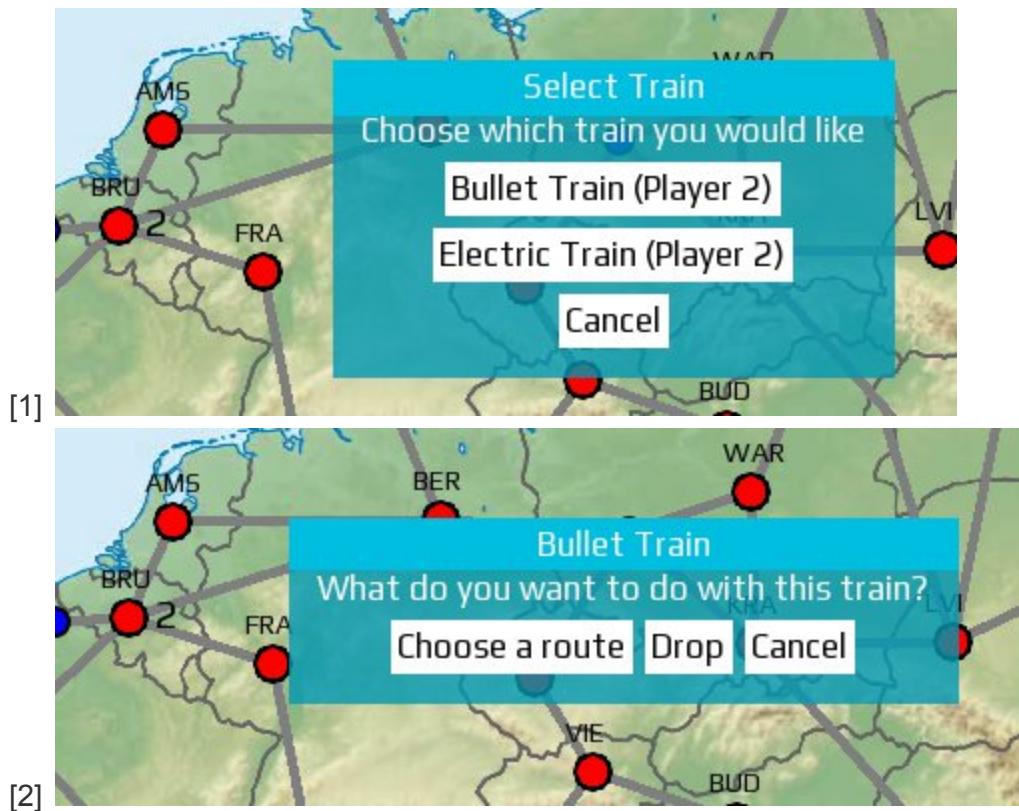
Bug ID	Bug Description
3	<p>Game ends prematurely with large score</p> <p>As highlighted by <i>Bug 2</i>, when a player achieves a large score, the game ends prematurely with that player as the winner:</p>

Resolution

The score ceiling of 10,000 points, of which we were previously unaware, was removed.

Bug ID	Bug Description
4	<p>Multiple trains at station replay bug</p> <p>Unable to replay the selection of a train where there are multiple trains at a station, game crashes</p>

e.g. Crashes on transition from [1] to [2] in replay mode

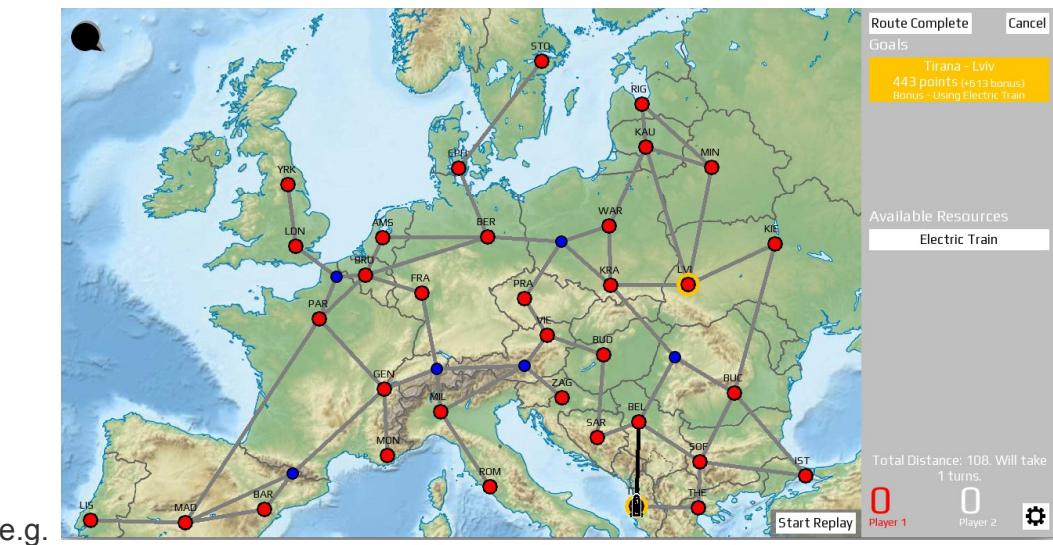


Resolution

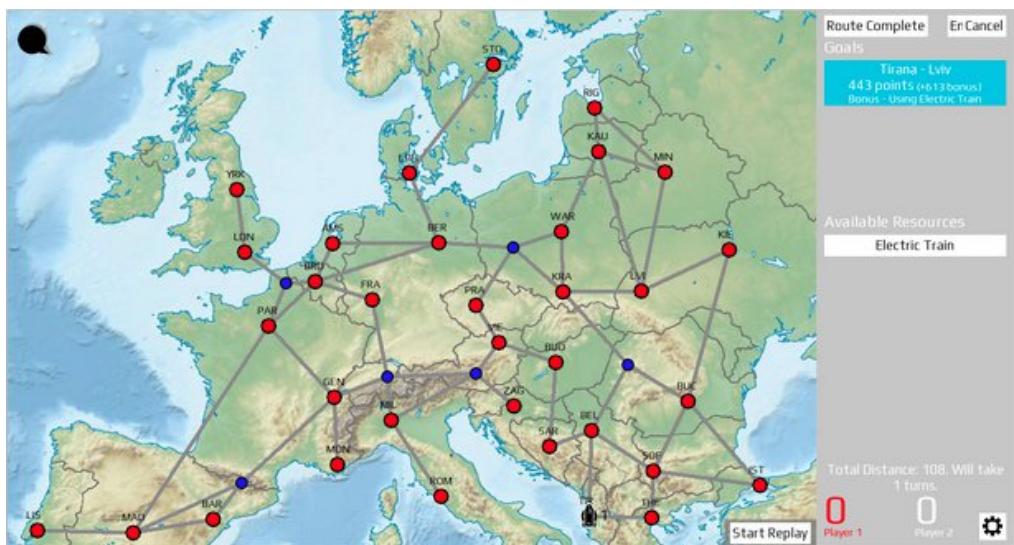
The multi-train dialog attaches an instance of a Train to each button, this train instance is then used to launch the next dialog which asks if you want to route/drop the train. Because the replay screen creates a new instance of every resource, storing a reference to the instance when the user clicked a button is not possible (because the instance will not exist in the replay).

To solve this, the DialogStationMultitrain class passes a string with information about the Train; its position, the player who owns it and its name. We store this string in the ReplayManager, and then we can use this to search through the resources in the Replay to find the correct instance.

Bug ID	Bug Description
5	Escape button while routing bug Pressing escape while in routing mode exists routing mode but without cancelling the set out route. The 'Route Complete' and 'Cancel' buttons don't disappear yet an 'End Turn' button appears (partially underneath the cancel button):



(Escape pressed)



(The train then progresses to Belgrade as routing wasn't cancelled)

Resolution

Cancel buttons were added where missing, so that the Escape key shortcut was no longer necessary. The escape key shortcut was then totally removed. This was a trade-off between the convenience of having the shortcut and removing the bugs caused by it, and we felt that the removal of the bugs was more important.

Appendix A - Unit Test Code

A.1 SettingsTest.java

```
package test;

import static org.junit.Assert.*;
import fvs.taxe.controller.SoundController;

import org.junit.After;
import org.junit.Before;
import org.junit.Test;

import com.badlogic.gdx.Gdx;

public class SettingsTest extends LibGdxTest{

    private SoundController soundController;
    private String currentSettings = null;

    @Before
    public void setUpSettings() {
        if(Gdx.files.local("taxesettings.json").exists()) {
            currentSettings = Gdx.files.local("taxesettings.json").readString();
            Gdx.files.local("taxesettings.json").delete();
        }
        soundController = new SoundController(false);
    }

    @Test
    public void defaultVolumeTest() {
        assertTrue(soundController.getMusicVolume() == 0.2f);
        assertTrue(soundController.getSoundVolume() == 1.0f);
    }

    @Test
    public void saveSettingsAndLoadTest() {
        float soundVolume = 0.5f, musicVolume = 0.5f;
        soundController.setSoundVolume(soundVolume);
        soundController.setMusicVolume(musicVolume);
        soundController.saveSettings();
        soundController = null;
        soundController = new SoundController(false);
        assertTrue(soundController.getMusicVolume() == musicVolume);
        assertTrue(soundController.getSoundVolume() == soundVolume);
    }

    @After
    public void afterTest() {
        if(currentSettings != null) {
            Gdx.files.local("taxesettings.json").writeString(currentSettings, false);
        }
    }
}
```

Appendix B - User and System Requirements

User Requirements

Requirement Number	Requirement Name
1	The game must be a two player game
2	The game is turn based, and each turn a player is given a new goal and new resources
3	The game ends when a certain score is achieved and the same number of turns have been completed by each player
4	The game must have a GUI, showing the routes, warnings, goals, start a new game, display some winner at the end
5	Game map must have some complexity
6	Player must be able to “describe” routes
7	Goals and Resources must be capped for each player
8	The player must be able to replay a set of turns
9	The player must be able to add and remove bits of the track while the game is being played

System Requirements

Requirement Number	Requirement Name
1.1	Must manage two sets of game state, e.g. the states of players' goals and resources etc.
1.2	Must have some way of notifying the user that the turn is over and then displaying the game state for the next player
1.3	Both players will be using the same machine
2.1	Have some way of determining when a turn is over, e.g a player can perform 3 actions per turn
2.2	Present the player a new goal each turn if they are not at the goal limit
2.3	Present the player 2 new resources every turn. The player has the option to accept/decline the new resources or swap them with the current resources.
3.1	Award score after completion of a goal
3.2	Score should relate to goal difficulty

3.3	Each player should be given a range of difficulties of goals, some completable perhaps within the same turn, others should take longer
3.4	Use score to calculate winner at the end of gameplay
3.5	There must be at least 10 different goals that players can be given
3.6	A player may have a maximum of 3 goals at any time
3.7	The game is a draw if the players finish with an equal score
4.1	Player either has some means of pushing a "start game" button or the game will start immediately when the game is launched
4.2	Must update when the turn ends to show the current player's goals, score and warnings
4.3	Warnings must be displayed when collisions or other events, e.g. someone jumped in front of the train, are about to occur
5.1	Must be at least two intersections, to allow for possibility for collisions
5.2	Must be at least five cities
5.3	Must be at least two routes between each city, e.g. a linear train line linking five cities would not suffice
5.4	There must be some city which is difficult to reach, which could be used for more difficult goals
5.5	The game must include at least 2 obstacles, one of which is a junction failure
6.1	Player must be able to choose a route to send a train on
6.2	Player should be able to do something about the warnings they have seen, and in the case of a collision, they must be able to take some action to stop it, e.g. stop a train or re-route a train
6.3	Already specified routings must run in a random order
7.1	The game must include at least 10 different attainable resources, e.g. trains and power-ups
7.2	A player may have a maximum of 7 resources at any time
8.1	The replay must be able to be played in "faster than real time"
8.2	The player has the option of selecting a set of turns to replay
9.1	Player can be allocated a "track/connection modifier" resource
9.2	Must have a mechanism for choosing which track to remove or where to add a new connection (length of the connection should be limited to avoid the creation of massive shortcuts)
9.3	Connection can't be removed if any train has it on its current route