**Load Feature:**

Feature: Load

As a user I want to load a saved game

Scenario: Player loads previously saved game # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/load.feature:4

Given player is in the Load page

And there is a saved session in the list

When Player chooses the saved session

And Player press the ?ok? button

Then desired session should be loaded by the system

1 Scenarios (1 undefined)

5 Steps (5 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^player is in the Load page$")

public void player\_is\_in\_the\_Load\_page() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^there is a saved session in the list$")

public void there\_is\_a\_saved\_session\_in\_the\_list() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player chooses the saved session$")

public void player\_chooses\_the\_saved\_session() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player press the ?ok? button$")

public void player\_press\_the\_ok\_button() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^desired session should be loaded by the system$")

public void desired\_session\_should\_be\_loaded\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Log Feature:**

Feature: Login

As a user I want to login with my account

Scenario: Player logins to the system # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/log.feature:4

Given players in the Login page

When Player enters the correct user credentials

And Player presses ?ok? button

Then player should be logged in to the system

1 Scenarios (1 undefined)

4 Steps (4 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^players in the Login page$")

public void players\_in\_the\_Login\_page() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player enters the correct user credentials$")

public void player\_enters\_the\_correct\_user\_credentials() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player presses ?ok? button$")

public void player\_presses\_ok\_button() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^player should be logged in to the system$")

public void player\_should\_be\_logged\_in\_to\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Menu Feature:**

Feature: Menu navigation

As a user I want to navigate between menu buttons

Scenario: Player clicks on desired button # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/menu.feature:4

Given player is in the menu

And Menu buttons are displayed

When Player touches a button

Then expected screen should be shown by the system

1 Scenarios (1 undefined)

4 Steps (4 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^player is in the menu$")

public void player\_is\_in\_the\_menu() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^Menu buttons are displayed$")

public void menu\_buttons\_are\_displayed() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player touches a button$")

public void player\_touches\_a\_button() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^expected screen should be shown by the system$")

public void expected\_screen\_should\_be\_shown\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Move Feature:**

Feature: Move

As a user I want to move the dog

Scenario: Player makes a movement in the game # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/move.feature:4

Given player is in the game screen

And Dog exist in the screen

When Player touches the screen

Then dog should be moved in desired direction by the system

1 Scenarios (1 undefined)

4 Steps (4 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^player is in the game screen$")

public void player\_is\_in\_the\_game\_screen() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^Dog exist in the screen$")

public void dog\_exist\_in\_the\_screen() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player touches the screen$")

public void player\_touches\_the\_screen() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^dog should be moved in desired direction by the system$")

public void dog\_should\_be\_moved\_in\_desired\_direction\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Register Feature:**

Feature: Register

As a user I want to create a profile

Scenario: player creates a new account # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/register.feature:4

Given Player is in the Register page

When player enters the user credentials

And player press the ?ok? button

Then New account should be opened by the system

1 Scenarios (1 undefined)

4 Steps (4 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^Player is in the Register page$")

public void player\_is\_in\_the\_Register\_page() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^player enters the user credentials$")

public void player\_enters\_the\_user\_credentials() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^player press the ?ok? button$")

public void player\_press\_the\_ok\_button() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^New account should be opened by the system$")

public void new\_account\_should\_be\_opened\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Save Feature:**

Feature: Save

As a user I want to save my game to continue later

Scenario: Player saves the current game session # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/save.feature:4

Given Player is in the game

And current game state has changed

When Player opens up the save menu

And Chooses for a suitable slot

Then Current game state should be saved by the system

1 Scenarios (1 undefined)

5 Steps (5 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^Player is in the game$")

public void player\_is\_in\_the\_game() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^current game state has changed$")

public void current\_game\_state\_has\_changed() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player opens up the save menu$")

public void player\_opens\_up\_the\_save\_menu() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Chooses for a suitable slot$")

public void chooses\_for\_a\_suitable\_slot() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^Current game state should be saved by the system$")

public void current\_game\_state\_should\_be\_saved\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**High Score Feature:**

Feature: High score

As a user I want to score up high score

Scenario: Player beats previous high scrore # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/score.feature:4

Given there is a previous high score

And Player is in the mode with high score

When Player makes a new record in the game

Then new high score should be saved by the system

1 Scenarios (1 undefined)

4 Steps (4 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^there is a previous high score$")

public void there\_is\_a\_previous\_high\_score() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^Player is in the mode with high score$")

public void player\_is\_in\_the\_mode\_with\_high\_score() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player makes a new record in the game$")

public void player\_makes\_a\_new\_record\_in\_the\_game() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^new high score should be saved by the system$")

public void new\_high\_score\_should\_be\_saved\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Server Feature:**

Feature: Connect to Server

As a user I want to play multiplayer game

Scenario: Player connects multiplayer game # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/server.feature:4

Given player logged in

And player is connected to the inernet

And player is in the lobby screen

When player chooses the server

And player press the ?ok? button

Then Connection should be supply by the system

1 Scenarios (1 undefined)

6 Steps (6 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^player logged in$")

public void player\_logged\_in() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^player is connected to the inernet$")

public void player\_is\_connected\_to\_the\_inernet() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Given("^player is in the lobby screen$")

public void player\_is\_in\_the\_lobby\_screen() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^player chooses the server$")

public void player\_chooses\_the\_server() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^player press the ?ok? button$")

public void player\_press\_the\_ok\_button() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^Connection should be supply by the system$")

public void connection\_should\_be\_supply\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

**Sound Feature:**

Feature: Set sound level

As a user I want adjust sound level

Scenario: Player changes sound level of the game # C:/Users/Mert/Desktop/Cucumber/CucumberProject/src/cucumber/features/sound.feature:4

Given player is in the Options page

When Player adjusted the sound level bar as desired

And Player press the ?done? button

Then sound level should be changed by the system

1 Scenarios (1 undefined)

4 Steps (4 undefined)

0m0,000s

You can implement missing steps with the snippets below:

@Given("^player is in the Options page$")

public void player\_is\_in\_the\_Options\_page() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player adjusted the sound level bar as desired$")

public void player\_adjusted\_the\_sound\_level\_bar\_as\_desired() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@When("^Player press the ?done? button$")

public void player\_press\_the\_done\_button() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}

@Then("^sound level should be changed by the system$")

public void sound\_level\_should\_be\_changed\_by\_the\_system() throws Throwable {

// Write code here that turns the phrase above into concrete actions

throw new PendingException();

}