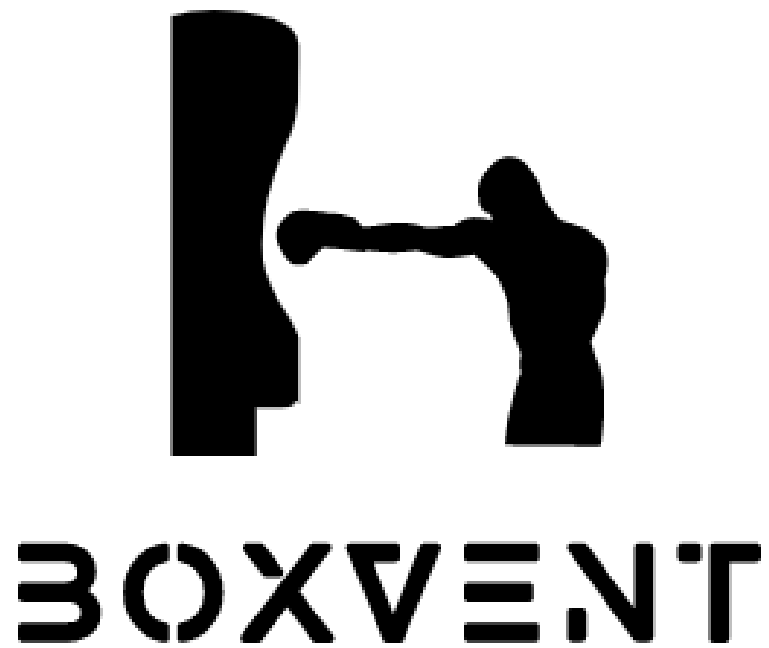


Test Plan



Vlad Dumitru – 3367231

Student at Fontys University of Applied Sciences – S3CB04

Sprint 3

Document version 2.0

Table of Contents

User Acceptance Tests	3
Verify that the create event button is not visible when the user doesn't have the required role.....	3
Verify that the create event button is visible when the user does have the required role	3
Verify that a fighter cannot be created without a boxing record.....	3
Verify than an event cannot be created with a date that is in the past	3
Verify than two events cannot be created at the same location at the same time	4
Verify that the event organizer cannot create events with no/negative tickets.....	4
Verify that the user doesn't see the navigation elements shown for the admin or event organizer	5
Verify that the event organizer doesn't see the navigation elements shown for the admin.....	5
Verify that a user or an organizer cannot delete a fighter	5
Verify that an admin or an organizer cannot create an event with a blank line up	5
Test Strategy	6

User Acceptance Tests

We will be conducting user acceptance tests to ensure that our customers will have a pleasant experience while navigating the website as well as checking how intuitive it is.

Verify that the create event button is not visible when the user doesn't have the required role

Step 1: User without admin or organizer role connects to the website.

Expected results: No option is available for creating an event.

Verify that the create event button is visible when the user does have the required role

Step 1: User with admin or organizer role connects to the website.

Step 2: User clicks on "Create event" button.

Expected results: The option is available for creating an event.

Verify that a fighter cannot be created without a boxing record

Step 1: User with admin role connects to the website.

Step 2: User navigates to the admin panel.

Step 3: User clicks on "Create fighter" button.

Step 4: User insert the boxers name but leaves the boxing record fields empty.

Step 5: User clicks the submit button

Expected results: The website doesn't make any API call and notifies the user that the record is missing.

Verify than an event cannot be created with a date that is in the past

Step 1: User with admin or organizer role connects to the website.

Step 2: User navigates to the admin panel.

Step 3: User clicks on "Create event" button.

Step 4: User creates the boxing line up.

Step 5: User picks a date that is in the past.

Step 6: User picks number of tickets.

Step 7: User picks ticket price.

Step 8: User submits the event.

Expected results: The website doesn't make any API call and notifies the user that the date is in the past/wrong format.

Verify than two events cannot be created at the same location at the same time

Step 1: User with admin or organizer role connects to the website.

Step 2: User navigates to the admin panel.

Step 3: User clicks on "Create event" button.

Step 4: User creates the boxing line up.

Step 5: User picks a date for the event.

Step 6: User picks number of tickets.

Step 7: User picks ticket price.

Step 8: User submits the event.

Expected results: An event gets created and added to the database. The user is informed that the creation process has been a success.

Step 9: User clicks on "Create event" button again.

Step 10: User creates the boxing line up.

Step 11: User picks the same date he picked for the previously created event.

Step 12: User submits the event.

Expected results: The website doesn't make any API call and notifies the user that there already exists an event at the chosen location and time.

Verify that the event organizer cannot create events with no/negative tickets

Step 1: User with admin or organizer role connects to the website.

Step 2: User navigates to the admin panel.

Step 3: User clicks on "Create event" button.

Step 4: User creates the boxing line up.

Step 5: User picks a date for the event.

Step 6: User picks 0 or negative number of tickets.

Step 7: User picks ticket price.

Step 8: User submits the event.

Expected results: The website doesn't make any API call and notifies the user that an event cannot be created with no tickets.

Verify that the user doesn't see the navigation elements shown for the admin or event organizer

Step 1: A user without admin or event organizer permission logs in to the website.

Expected results: The user does not see the navigation links for the admin or organizer.

Verify that the event organizer doesn't see the navigation elements shown for the admin

Step 1: A user with event organizer permission logs in to the website.

Expected results: The user does not see the navigation links for the admin but does see them for organizer.

Verify that a user or an organizer cannot delete a fighter

Step 1: A user with or without event organizer permission logs in to the website.

Step 2: The user navigates to the fighters page.

Expected results: There is no "Remove" button shown next to the fighters.

Verify that a user or an organizer cannot delete an event

Step 1: A user with or without event organizer permission logs in to the website.

Step 2: The user navigates to the events page.

Expected results: There is no "Remove" button shown next to the events.

Verify that an admin or an organizer cannot create an event with a blank line up

Step 1: User with admin or organizer role connects to the website.

Step 2: User navigates to the admin panel.

Step 3: User clicks on "Create event" button.

Step 4: User creates the boxing line up with no fighters.

Step 5: User picks a date for the event.

Step 6: User picks the number of tickets.

Step 7: User picks ticket price.

Step 8: User submits the event.

Expected results: The website doesn't make any API call and notifies the user that an event cannot be created with no fighters.

Test Strategy

What will be tested?

For our backend we will be testing most components by using unit tests and integration tests. We will be using unit tests for testing individual components, and we will be using integration tests to test how certain components integrate with each other.

We will also be testing the whole website by doing the previously mention user acceptance tests, by following the required steps to see how a normal customer of the website would react to them and if everything works accordingly.

Why will it be tested?

These tests will be performed with the scope of ensuring the continuous development of the website as well as assisting in the development process by catching bugs and errors faster, making the development process contain less mistakes.