## "Eventures" Web App

You are given an **Web app** Eventures

**You are already familiar with the app from QA Basics Course, but go ahead and Look below** to understand **how the app works**.

Graphical user interface, text, application, email

Description automatically generated

The **Web app** holds a list of **events**, created by different **users** (after register / login).

### Users

* Users can **register** on the site with their own provided username, email, password, first name and last name.
* Users can **login** by username + password.
* After either **register / login** users can **view** existing events, **edit** / **delete** owned events, and **create** new events.

### Events

Each **event** has a name, place, start date, end date, total tickets, price per ticket and owner.

The **event owner** is the user who created a certain event.

* Anyone can view all events but editing and deleting events is limited to their respective owners.
* By requirement, the **event owner can edit and delete his own events** but **cannot edit and delete other user's events**.

## Key Features of Eventures Web App

### Register

Users can **register** on the site with their own provided **username**, **email**, **password**, **first name** and **last name**:

Graphical user interface, application

Description automatically generated

### Graphical user interface, text, application, email Description automatically generatedLogin

Users can **login** by username + password:

**Requirements:**

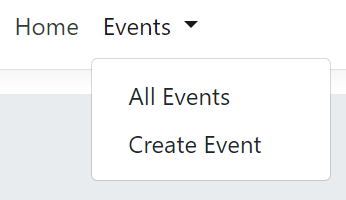
* Users **can't log in** without either a **username** or **password**.
* You can log in with **user: guest, password: guest**.
* When clicked **Login button redirects** to **Home page**.

### Homepage

**After** either **register / login**, users are taken to the **Home page**.Graphical user interface, text, application, email

Description automatically generated

**Requirements:**

* Users should have **two ways to navigate** to **all events** page: clicking **"all events" link** and from the **drop-down menu** under "Events".
* User should have **two ways to navigate** to **Create event**: clicking **"new event"** link and from the **drop-down menu** under "Events".
* Users should be able to **log out**, by clicking the **Logout button**.
* At the home page, the **number of total events** in the system is shown
* At the home page, the **number of user’s owned events** are shown.

### Create New Event

**Logged users** can **create new** **events**.

Graphical user interface, text, application, email

Description automatically generated

**Requirements:**

* **Each event** should **have a name**.
* Each **event** should **have a place**.
* Each **event** should have a **Start** and **End date**.
* **Start date** should be **before** the **End date**.
* **Start date** should be in the **future**.
* The **year** of the **start date** should be **before 2100**.
* **End date** should be **after the start date.**
* **Total Tickets** maximum should be **1000**.
* **Price Per Ticket** should be **<= 1000**
* When clicked **Create event button, creates** the **new event** and **redirects** user to **All events page,** with the newly created event.
* **Back to list** button**,** when clicked, redirects to **All events page**.

### All Events

Users can **view** all existing events but can **edit** and **delete** only events that **they created**.

**Graphical user interface, text, application

Description automatically generated**

**Requirements:**

* There **are Delete | Edit** links for the event that the **current user owns**.
* There **are not Delete | Edit** links for the event that is **owned by another user**.
* There is a **Create new link,** which leads to creation of new event form.

### Edit Event

**Logged user** can **edit** the **name**, **place**, **start date**, **end date**, **total tickets**, **and price** per ticket **for his own events**.

**Graphical user interface, text, application, email

Description automatically generated**

**Requirements:**

* All fields are **editable**.
* When clicked **Edit button**, redirects to All events page, where edited **event is saved**.
* **Back to List button** leads to All events page, **without saving the changes** made to the event.

### Delete Event

From the All events page**, logged user** can delete his own events. When clicked, the [Delete] link leads to **Delete Existing event form**.

**Graphical user interface, application

Description automatically generated**

**Requirements:**

* **Name field** is **not editable** and is populated with the name of the event for deletion.
* **Place field** is **not editable** and is populated with the place of the event for deletion.
* **Delete button**, when clicked, deletes the event, and redirects to **All events page**, where the **deleted event is not present.**
* **Back to List button**, when clicked, redirects to **All events page**, **without deleting the event**.

# **Test Scenarios and Test Cases**

## **Test Cases for User:**

#### Test Case 1: Test\_User\_Register

Description: Verify that a new user can successfully register on the Eventures web application.

Steps:

Go to the "Registration" page on the Eventures web application.

Fill in the registration form with valid data (username, email, password, first name, and last name).

Click on the "Register" button.

Expected Results: The user is successfully registered, and a welcome message containing the username is displayed on the Home page.

#### Test Case 2: Test\_User\_Login

Description: Verify that a registered user can log in to the Eventures web application.

Steps:

Go to the "Login" page on the Eventures web application.

Fill in the login form with valid credentials (username and password).

Click on the "Log in" button.

Expected Results: The user is successfully logged in, and a welcome message containing the username is displayed on the Home page.

#### Test Case 3: Test\_User\_Logout

Description: Verify that a logged-in user can successfully log out from the Eventures web application.

Steps:

Go to the Home page on the Eventures web application.

Click on the "Logout" button.

Expected Results: The user is successfully logged out, and the Home page displays the main page content without any user-specific information.

#### Test Case 4: Test\_HomePage\_LoginPageLink\_InNavigation

Description: Verify that the "Log in" page can be accessed via the navigation bar on the Home page.

Steps:

Go to the Home page on the Eventures web application.

Click on the "Log in" link in the navigation bar.

Expected Results: The user is redirected to the "Log in" page, and the Log in page content is displayed.

#### Test Case 5: Test\_HomePage\_LoginPageLink\_OnPage

Description: Verify that the "Log in" page can be accessed via the link on the main page.

Steps:

Go to the Home page on the Eventures web application.

Click on the "Log in" link on the main page.

Expected Results: The user is redirected to the "Log in" page, and the Log in page content is displayed.

#### Test Case 6: Test\_HomePage\_RegisterPageLink\_InNavigation

Description: Verify that the "Register" page can be accessed via the navigation bar on the Home page.

Steps:

Go to the Home page on the Eventures web application.

Click on the "Register" link in the navigation bar.

Expected Results: The user is redirected to the "Register" page, and the Register page content is displayed.

#### Test Case 7: Test\_HomePage\_RegisterPageLink\_OnPage

Description: Verify that the "Register" page can be accessed via the link on the main page.

Steps:

Go to the Home page on the Eventures web application.

Click on the "Register" link on the main page.

Expected Results: The user is redirected to the "Register" page, and the Register page content is displayed.

## Test Cases for Events:

#### Test Case 8: Test\_HomePage\_AllEventsLink

**Description:** Verify that the "All Events" link on the Home page redirects to the "All Events" page.

**Steps:**

Go to the Home page on the Eventures web application.

Click on the "All Events" link on the main page.

**Expected Results:** The user is redirected to the "All Events" page, and the page displays the "All Events" header and a "Create New" link.

#### Test Case 9: Test\_AllEventsPage\_ThroughNavigation

**Description:** Verify that the "All Events" page can be accessed through navigation from the Home page.

**Steps:**

Go to the Home page on the Eventures web application.

Click on the dropdown menu.

Click on the "All Events" link from the dropdown menu.

**Expected Results:** The user is redirected to the "All Events" page, and the page displays the "All Events" header and a "Create New" link.

#### Test Case 10: Test\_HomePage\_CreateEventPageLink

**Description:** Verify that the "Create Event" link on the Home page redirects to the "Create Event" page.

**Steps:**

Go to the Home page on the Eventures web application.

Click on the "Create Event" link on the main page.

**Expected Results:** The user is redirected to the "Create Event" page, and the page displays the "Create New Event" header and a "Back to List" link.

#### Test Case 11: Test\_CreateEventPage\_ThroughNavigation

**Description:** Verify that the "Create Event" page can be accessed through navigation from the Home page.

**Steps:**

Go to the Home page on the Eventures web application.

Click on the dropdown menu.

Click on the "Create Event" link from the dropdown menu.

**Expected Results:** The user is redirected to the "Create Event" page, and the page displays the "Create New Event" header and a "Back to List" link.

#### Test Case 12: Test\_CreateEventPage\_BackToListLink

**Description:** Verify that the "Back to List" link on the "Create Event" page redirects to the "All Events" page.

**Steps:**

Go to the "Create Event" page on the Eventures web application.

Click on the "Back to List" link.

**Expected Results:** The user is redirected to the "All Events" page, and the page displays the "All Events" header.

#### Test Case 13: Test\_CreateEventPage\_CreateEvent\_ValidData

**Description:** Verify that a new event can be created with valid data on the "Create Event" page.

**Steps:**

Go to the "Create Event" page on the Eventures web application.

Fill in the event details (name, place, total tickets, and price per ticket).

Click on the "Create" button.

**Expected Results:** The user is redirected to the "All Events" page, and the newly created event is displayed with the "Delete" and "Edit" buttons.

#### Test Case 14: Test\_CreateEventPage\_CreateEvent\_InvalidData

**Description:** Verify that an error message is displayed when trying to create an event with invalid data on the "Create Event" page.

**Steps:**

Go to the "Create Event" page on the Eventures web application.

Fill in the event details with invalid data (e.g., negative ticket number or price).

Click on the "Create" button.

**Expected Results:** The user remains on the "Create Event" page, and an error message indicating the invalid input is displayed.

#### Test Case 15: Test\_DeleteEvent

**Description:** Verify that an event can be successfully deleted from the "All Events" page.

**Steps:**

Go to the "All Events" page on the Eventures web application.

Click on the "Delete" button of the event to be deleted.

**Expected Results:** The user is redirected to the "All Events" page, and the deleted event is no longer displayed on the page.

#### Test Case 16: Test\_EditEvent\_ValidData

**Description:** Verify that an event can be successfully edited with valid data on the "Edit Event" page.

**Steps:**

Go to the "All Events" page on the Eventures web application.

Click on the "Edit" button of the event to be edited.

Modify the event details (e.g., change the event name).

Click on the "Save" button.

**Expected Results:** The user is redirected to the "All Events" page, and the edited event with the updated data is displayed.

#### Test Case 17: Test\_EditEvent\_InvalidData

**Description:** Verify that an error message is displayed when trying to edit an event with invalid data on the "Edit Event" page.

**Steps:**

Go to the "All Events" page on the Eventures web application.

Click on the "Edit" button of the event to be edited.

Modify the event details with invalid data (e.g., negative ticket number or price).

Click on the "Save" button.

**Expected Results:** The user remains on the "Edit Event" page, and an error message indicating the invalid input is displayed.