

NOVEMBER, 2022



# Test Plan

## version 2

For the individual project  
of Semester 3, ICT  
Software Engineering

Created by  
**Iulia Toderaşcu**

Class  
**S3-CB02**

# Table of Contents

**01** Introduction

**02** Objectives and tasks

**03** Scope

**04** Test Cases

**05** Test strategy

**06** User acceptance tests

# 1. Introduction

By implementing the idea of Evento I wanted to show the skills that I acquired until now and my ability to learn fast how to work with different frameworks. My focal points are implementing the practices that I wasn't too confident about, understanding them, and developing efficient solutions.

This document will contain all the information that my testing is based on. All the results of the test cases will be presented in the Test Report.

# 2.Objectives and tasks

## OBJECTIVES

- defining tests responsibilities
- provide instructions for the tests
- analyze the results
- improve features

## TASKS

In this part of my project, the tasks are clear: I want to offer clear instructions in order to observe the functionality of my app and the user experience. As so, my main tasks are:

- test conducting
- testing
- post-testing
- problem reporting

# 3.Scope

## **What is being tested**

- functional requirements
- non-functional requirements
- usability
- performance

## **Tactics: how are the functionalities going to be tested**

- In order to ensure that every aspect of my application was tested, I created test cases based on my User Requirements Specification document. In each test case, instructions will outline the actions that must be taken to complete the task.

# Test cases

## Test case no.1 Log in - unsuccessful

- **Instructions:**
  - 1.Add a random email
  - 2.Add a random password
  - 3.Try to log in to the platform
- **Expected result:**

Cannot log it. The page will show an error message.

## Test case no.2 Log in - successful

- **Instructions:**
  - 1.Add email "test@gmail.com"
  - 2.Add password "test123"
  - 3.Try to log in to the platform
- **Expected result:**

Can log in. The user will be redirected to the home page and will have client permissions.

# Test cases

## Test case no.3 Register - client

- **Instructions:**

1. Navigate to the "Register" page
2. Add your credentials: name, email, password
3. Choose "Client" for the type of account
4. Try to log in.

- **Expected result:**

Can log in. The user is redirected to the home page, and has client permissions

## Test case no.3 Register - organizer

- **Instructions:**

- Navigate to the "Register" page
  - Add your credentials: name, email, password, business details, type of events, payment etc.
  - Choose "Organizer" for the type of account
  - Try to log in.
- 1.

- **Expected result:**

Can log in. The user is redirected to the home page, and has organizers permissions

# Test cases

## Test case no.4 Create event -successful

- **Instructions:**

1. Navigate to the "Log in" page
2. Add email "organizer@gmail.com"
3. Add password "abc1"
4. Log in and navigate to the "Create event" page
5. Fill in the details necessary for creating a new event (name, type of event, location, time, date, number of available tickets)
6. Confirm the changes only when all fields are filled
7. Navigate to the "My events" page

- **Expected result:**

The created event appears on the website, becomes available to all users



# Test cases

## Test case no.5 Create event -unsuccessful

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "organizer@gmail.com"
- 3.Add password "abc1"
- 4.Log in and navigate to the "Edit event" page
- 5.Chose one of the events available and activate editing mode
- 6.Change any field you want.
- 7.Confirm
- 8.Navigate to "My events" page
- 9.Go to the event you just edited

- **Expected result:**

Can edit the details about an already existing event.

# Test cases

## Test case no.6 Delete an event

- **Instructions:**

1. Navigate to the "Log in" page
2. Add email "organizer@gmail.com"
3. Add password "abc1"
4. Log in and navigate to the "Edit event" page
5. Chose one of the events available and chose to delete it
6. Confirm
7. Navigate to the "My events" page

- **Expected result:**

The deleted event isn't shown anymore on the website

# Test cases

## Test case no.7 See statistics about Events

- **Instructions:**

1. Navigate to the "Log in" page
2. Add email "organizer@gmail.com"
3. Add password "abc1"
4. Log in and navigate to the "Statistics" page

- **Expected result:**

The user will be able to see all the automatic generated statistics about the sales

# Test cases

Test case no.8 See available events in the platform

- **Instructions:**

1. Navigate to the "Log in" page
2. Add email "client@gmail.com"
3. Add password "abc2"
4. Log in to the platform
5. Navigate to the "All events" page

- **Expected result:**

The user can see all the available events and details about them

# Test cases

## Test case no.9 Buy tickets

- **Instructions:**

1. Navigate to the "Log in" page
2. Add email "client@gmail.com"
3. Add password "abc2"
4. Log in to the platform
5. Navigate to the "All events" page
6. Choose an available event
7. Select the number of tickets wanted
8. Confirm the purchase.
9. Navigate to "Profile" page

- **Expected result:**

The purchased ticket is visible on the Profile page

# Test cases

## Test case no.10 Filter events

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "client@gmail.com"
- 3.Add password "abc2"
- 4.Log in to the platform
- 5.Navigate to the "All events" page
- 6.Choose the "music concerts" filter
- 7.Confirm

- **Expected result:**

Only the events with the tag "music concerts" are visible on the website

# Test cases

## Test case no.11 Subscribe to a type of event

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "client@gmail.com"
- 3.Add password "abc2"
- 4.Log in to the platform
- 5.Navigate to the "All events" page
- 6.Choose an event that you like
- 7.Subscribe to the event
- 8.Navigate to the Profile page

- **Expected result:**

All the subscribed events will be shown on the Profile, meaning that on the home page the user will receive a feed with the same category of events that he likes

# Test cases

## Test case no.12 Manage accounts

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "admin@gmail.com"
- 3.Add password "abc3"
- 4.Log in to the platform
- 5.Navigate to the "Organizers" page
- 6.Choose any organizer account

- **Expected result:**

The admin is able to see all organizer's accounts and the events organized by them



# Test cases

## Test case no.13 Delete organizer accounts

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "admin@gmail.com"
- 3.Add password "abc3"
- 4.Log in to the platform
- 5.Navigate to the "Organizers" page
- 6.Choose any organizer account
- 7.Delete the account

- **Expected result:**

The admin is able to delete the accounts of the organizer in case they are not respecting the legal terms

# Test cases

## Test case no.14 Buy ticket at an event with seating system

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "client@gmail.com"
- 3.Add password "abc1"
- 4.Log in to the platform
- 5.Navigate to the "All events" page
- 6.Choose any available event
- 7.Chose the number of tickets that you want
- 8.Choose the desired seat(s)
- 9.Confirm the payment

- **Expected result:**

The purchased ticket will be visible on the Profile page and the seat(s) will be marked as occupied and won't be available to buy anymore

# Test cases

## Test case no.15 Receive suggestions for seats

- **Instructions:**

- 1.Navigate to the "Log in" page
- 2.Add email "client@gmail.com"
- 3.Add password "abc1"
- 4.Log in to the platform
- 5.Navigate to the "All events" page
- 6.Choose any available event
- 7.Chose the number of tickets that you want
- 8.Choose the desired suggestion for the seats
- 9.Confirm the payment

- **Expected result:**

The purchased tickets will be visible on the Profile page and the seats will be marked as occupied and won't be available to buy anymore

# 6. Test strategy

## CONTENT

- **why** should the application be tested
- **what** is tested
- **how** is it tested

## WHY

- Testing ensures the quality of the software by discovering bugs/defects before it is delivered to the client.
- This improves the reliability and usability of the software.
- Reliable and high-performance software is ensured by thoroughly tested software.
- Occasionally, we find that a fully developed software product does not meet user requirements because it lacks some basic functionality. A mistake may have been made during the requirements gathering or coding phases. To fix such errors, we may have to start the development process over again. The process of fixing such mistakes becomes very tedious, time-consuming, and expensive. As a result, testing software during development is always desirable.

# WHAT

In my application, the following features should be tested:

- CRUD functionalities for **events** and **accounts**
- Logging and registration
- Authentication and authorization
- Real-time tickets stock status
- Creating a QR code for the purchase
- Sending the QR code by email

# HOW

After researching, I decided to choose the following test methods for my application:

- Unit testing

Unit testing ensures that each part of a program works properly on its own, thereby increasing testing strategies and reducing wasted time.

- White box testing

White box testing involves testing the product's underlying structure, architecture, and code to validate input-output flow and enhance the design, usability, and security.

- Security testing

To ensure that the software system and applications are secure, security testing uncovers the vulnerabilities of the system. In these tests, any potential flaws or weaknesses in the software system will be studied in order to find out if they could lead to the loss of data, revenue, or reputation of the company or its employees.

- Performance testing

Performance testing examines the speed, stability, reliability, scalability, and resource usage of a software application under a specified workload.

- Integration testing

An integrated system's compliance with a set of requirements is ensured through integration testing. To ensure proper system operation, it is carried out in a hardware and software environment that is integrated.

- Security testing

To ensure that the software system and applications are secure, security testing uncovers the vulnerabilities of the system. In these tests, any potential flaws or weaknesses in the software system will be studied in order to find out if they could lead to the loss of data, revenue, or reputation of the company or its employees.

Functionality	Tetsing method
CRUD functionalities	Unit testing, integration testing
Logging and registration	Unit testing, Integration testing, Security testing
Authentification and authorization	Unit testing, security testing
Real-time tickets stock status	Unit testing, performance testing
Creating a QR code for the purchase	Unit testing, performance testing
Sending the QR code by email	Unit testing, security testing, performance testing

# Questions? Contact me.

Student e-mail: 475741@student.fontys.nl

Personal e-mail: toderacuiulia@gmail.com

Phone number: +40757655970

