

SEPTEMBER, 2022



User Requirements Specification

For the individual project
of Semester 3, ICT
Software Engineering

Created by
Iulia Toderaşcu

Class
S3-CB02

Table of Contents

01 Functional requirements

02 Non-functional requirements

03 Moscow Method

04 Use cases

05 User stories & Acceptance criteria

1. Functional requirements

FR-01: Support registering users

Interested users can visit the website to learn more about it. They can register on the website as clients/organizers.

FR-02: Manage accounts

The admin can manage the organizer's accounts in order to have only verified events on the website

FR-03: Manage events

The events must be managed by their organizers (CRUD operations)

FR-04: Support filtering events

Each event will belong to a category(music concerts, online events, theatre, sports events etc.) so it can provide smooth navigation on the website

FR-05: Subscribe to a specific type of events/organizers

The clients can subscribe to categories of events preferred or certain event organizers in order to customize their main page with information based on their preferences.

FR-06: Organizers have access to statistics about their events

Organizers can see statistics about the sales and the most successful events in general

FR-07: Support showing event information

It should be possible for every logged-in user to find out information about a particular event

FR-08: Support buying tickets

The logged-in clients can buy tickets to the available events.

FR-09: Generate client and organizer profile

Each client/organizer will have his/her own profile containing information about his/her account details and preferences

FR-10: Give suggestions for seats

For the events that have a seating system, the clients will receive suggestions for the seats according to the number of places desired

2.Non-functional requirements

- Speed (optimize the application's speed taking into account the database service)
- Security (create a system that protects sensitive data)
- Usability (easy navigation, useful purpose of features)

Moscow method

Must-have

- login system
- different permissions based on the type of account
- show event information
- a system that allows organizers to manage events
- a system that will enable clients to buy tickets to the events

Should-have

- intuitive design
- a system that can be accessed from multiple devices simultaneously
- an efficient way of storing data about events and users

Could-have

- mobile version
- QR codes with the tickets sent to the email addresses

Won't-have

- payment system
- different memberships for users

Use cases

Use case no.1 Log in FR-01

- **Actor:** user (no matter the permission)
- **Main Success Scenario:**
 1. User navigates to the "Log in" page from the bar menu
 2. The user enters his password.
 3. The user enters his email.
 4. The system validates the credentials and identifies the type of account based on them(client, organizer, admin)
 5. The system confirms that the user has logged in successfully
 6. The user has now access to the functionalities that his type of account is providing.
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.2 Register FR-01

- **Actor:** user (no matter the permission)
- **Main Success Scenario:**
 1. User navigates to the "Register" page from the bar menu
 2. The user sets his password and email.
 3. The user chooses the type of account that he wants.
 4. The system validates the credentials and creates the type of account based on the user's choice.
 5. The system confirms that the user has logged in successfully
 6. The user has no access to the functionalities that his type of account is providing.
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no. 2.2 Register as an organizer FR-01

- **Actor:** user (no matter the permission)
- **Main Success Scenario:**
 1. User navigates to the "Register" page from the bar menu
 2. The user sets his password and email.
 3. The user chooses the type of account that he wants.
 4. The system validates the credentials and creates the type of account based on the user's choice.
 5. The system confirms that the user has logged in successfully
 6. The user has no access to the functionalities that his type of account is providing.
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.3 Create an event FR-3

- **Actor:** user with organizer permission
- **Precondition:** organizer is logged in
- **Main Success Scenario:**
 1. Organizer navigates to the "Create event" page from the bar menu
 2. Organizer sets up the details of the event: name, type of event, location, time, date, and number of available tickets.
 3. Organizer confirms the details introduced.
 4. The system validates the data and posts the event on the page
- **Extensions:**
 - 3a. Mandatory data not introduced (date, time, location, no. of tickets)**
 - .1: The system displays an error message indicating the blank spaces
 - .2: The event is not created
 - .2: End of use case.
 - 3b. Similar data already exists.**
 - .1: The system indicates the data that is similar to another existing event and shows an error message.
 - .2: End of use case.

Use cases

Use case no.4 Edit an event FR-3

- **Actor:** user with organizer permission
- **Precondition:** organizer is logged in, the event that has to be edited exists already
- **Main Success Scenario:**
 1. Organizer navigates to the "Edit event" page from the bar menu
 2. The organizer chooses the event that he/she wants to edit (only the events that the logged-in user organizes will be displayed).
 3. The organizer can edit the details about the event and the changes will be saved and posted on the website.
- **Extensions:**
 - 2a. Trying to edit a past event:
 - .1: The system displays an error message containing the date on which the event took place
 - .2: End of use case.

Use cases

Use case no.5 Delete an event FR-3

- **Actor:** user with organizer permission
- **Precondition:** organizer is logged in, the event that has to be edited exists already
- **Main Success Scenario:**
 1. User navigates to the "Edit event" page from the bar menu
 2. The organizer chooses the event that he/she wants to delete(only the events that the logged-in user organizes will be displayed).
 3. The organizer can delete the event but in case some tickets were sold already, a notification will be sent to the users that bought the ticket.
 4. The organizer has to write a message to the clients: either refund the money or the date of the event will be communicated in the future.
- **Extensions:**
 - 2a. Trying to edit a past event:
 - .1: The system displays an error message containing the date on which the event took place
 - .2: End of use case.

Use cases

Use case no.6 Generate statistics FR-6

- **Actor:** user with organizer permission
- **Precondition:** organizer is logged in and it has to have at least 2 past events with tickets sold
- **Main Success Scenario:**
 1. Organizer navigates to the "Statistics" page from the bar menu
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.7 View available events in the platform FR-7

- **Actor:** client
- **Precondition:** client is logged in
- **Main Success Scenario:**
 1. The client navigates to the "Home" page from the bar menu
 2. The client can see all the available events
 3. The client can navigate to the events page by clicking on the event
 4. The client can see all the information about the event and (if available) he can buy a ticket (see next Use Case)
- **Extensions:**
 - 2a: No available events
 - .1: The past events won't be shown, only the future ones. If there are no future events, the page will be empty.
 - .2: End of use case.

Use cases

Use case no.8 Buy ticket to event FR-8

- **Actor:** client
- **Precondition:** client is logged
- **Main Success Scenario:**
 1. The client navigates to the "Home" page from the bar menu
 2. The client can see all the available events
 3. The client can navigate to the events page by clicking on the event
 4. The client can select the number of tickets wanted
 5. The client confirms the payment.
- **Extensions:**
 - 4a. Each user can buy up to 10 tickets:
 - .1: The system displays an error message if the number of tickets is greater than 10.
 - .2: End of use case.

Use cases

Use case no.9 Apply filter to events FR-4

- **Actor:** client
- **Precondition:** client is logged
- **Main Success Scenario:**
 1. The client navigates to the "Home" page from the bar menu
 2. The client can see all the available events
 3. The client can choose a filter to apply to the results. The filter includes the different types of events: concerts, theatre, movies dance, online events etc.
- **Extensions:**
 - 3a. No future events with the type of filter chosen
 - .1: The system displays an error message and the result page will appear empty
 - .2: End of use case.

Use cases

Use case no.10 Subscribe to events FR-5

- **Actor:** client
- **Precondition:** client is logged
- **Main Success Scenario:**
 1. The client navigates to the "Home" page from the bar menu
 2. The client can see all the available events
 3. The client can navigate to the events page by clicking on the event
 4. The client can subscribe to a type of events/ specific organizer in order to receive first the relevant announcements
- **Extensions:**
 - 4a. Already subscribed to the type of event/organizer
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.11 Organizers accounts management FR-2

- **Actor:** admin
- **Precondition:** admin is logged in
- **Main Success Scenario:**
 1. The client navigates to the "Organizers" page from the bar menu
 2. The admin can see all the organizer's accounts
 3. The admin can navigate to the organizer's profile page
 4. The admin can delete the organizer's profile if it's the case
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.11.2 Delete account FR-2

- **Actor:** admin
- **Precondition:** admin is logged in
- **Main Success Scenario:**
 1. The client navigates to the "Organizers" page from the bar menu
 2. The admin can see all the organizer's accounts
 3. The admin can navigate to the organizer's profile page
 4. The admin can delete the organizer's profile if it's the case
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.12 Buy tickets to events with seating system FR-8

- **Actor:** client
- **Precondition:** client is logged
- **Main Success Scenario:**
 1. The client navigates to the "Home" page from the bar menu
 2. The client can see all the available events
 3. The client can navigate to the events page by clicking on the event
 4. The client can select the number of tickets wanted
 5. The client can select the seat(s) wanted.
 6. The client confirms the payment.
 7. The seats bought are marked as occupied.
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.13 Receive suggestions when buying tickets FR-10

- **Actor:** client
- **Precondition:** client is logged in and choose an event with a seating system to buy tickets
- **Main Success Scenario:**
 1. The client chooses the number of tickets that he/she wants to buy
 2. The system makes suggestions if the desired amount of tickets is greater than 2.
 3. The client can confirm the suggestion or he/she can choose their own seating pattern.
 4. The client confirms the payment.
 5. The seats bought are marked as occupied.
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

Use cases

Use case no.14 View profile FR-9

- **Actor:** client
- **Precondition:** client is logged
- **Main Success Scenario:**
 1. The client navigates to "My profile" page
 2. The client can see all the past events attended
- **Extensions:**
 - 4a. Invalid password with valid email:
 - .1: The system displays an error message
 - .2: End of use case.

User stories

User story no.1

- **As an** organizer using Evento.
- **I can** create/edit/delete events.
- **So that** I can manage my occasions.
- **Acceptance criteria:**
 - The events can be created
 - The events can be edited
 - The events can be deleted
 - The events can be displayed
 - The users can interact with the existing events
 - The events can be managed only by their organizer

User story no.2

- **As a** client using Evento.
- **I can** buy tickets to the available events.
- **So that** I can attend the events I want.
- **Acceptance criteria :**
 - The existing events can be displayed
 - The client can buy tickets to the event
 - The client can have a history of his/her purchases

User stories

User story no.3

- **As a** client using Evento.
- **I can** subscribe to the type of events I find interesting
- **So that** I can customize my feed.
- **Acceptance criteria:**
 - The existing events are organized by category
 - The client can subscribe to a type of event
 - The feed of the website can contain the newest events of the subscribed category

User story no.4

- **As an** admin using Evento.
- **I can** delete organizers' accounts if necessary.
- **So that** I can manage the content on my website.
- **Acceptance criteria:**
 - The organizers' accounts can be deleted only by the admin
 - The admin can see every active organizer account

User stories

User story no.5

- **As an** organizer using Evento.
- **I can** see statistics about the sales of tickets.
- **So that** I can improve my strategies.
- **Acceptance criteria:**
 - The data about the events is kept
 - The organizer can see statistics only about their events
 - The organizer can

User story no.6

- **As a** client using Evento.
- **I can** apply filters when I search for an event.
- **So that** I can find much more easily the events I want.
- **Acceptance criteria:**
 - The events belong to at least one category of the filter
 - The events can be sorted based on the filter chosen.

User stories

User story no.7

- **As a** client using Evento.
- **I can** choose my own seat if the event has a seating system
- **So that** I can save it for myself.
- **Acceptance criteria:**
 - The client can choose from the available seats
 - The seats bought are marked as occupied

User story no.8

- **As an** organizer using Evento.
- **I can** choose a seating system for my events.
- **So that** I can manage the number of people attending the event
- **Acceptance criteria:**
 - The organizers can implement the seating system for their events

User stories

User story no.9

- **As a** client using Evento.
- **I can** receive suggestions for seats when I chose the number of tickets that I want.
- **So that** I purchase faster.
- **Acceptance criteria:**
 - The client can choose the number of seats wanted.
 - The system can return suggestions for the number of seats desired.

User story no.10

- **As a** client using Evento.
- **I can** view my profile.
- **So that** I can see the events I attended and the future events I will attend.
- **Acceptance criteria:**
 - The clients have their own profile
 - The clients can view their profile

User stories

User story no.11

- **As a** client using Evento.
- **I can** view my profile.
- **So that** I can see the events I attended and the future events I will attend.
- **Acceptance criteria:**
 - The clients have their own profile
 - The clients can view their profile

Questions? Contact me.

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

Personal e-mail: toderacuiulia@gmail.com

Phone number: +40757655970

