



# **ESN Ticketing App**

## **Project Documentation**

Version 2.1.0

Trajkovska<sup>1</sup>, Milivojčević<sup>1</sup>, and Šoberl<sup>1</sup>

<sup>1</sup>University of Primorska

August 13, 2023

# Authors

Students: Karolina Trajkovska, Milan Milivojčević

Mentor: dr. Domen Šoberl *University of Primorska, Faculty of Mathematics, Natural Sciences and Information Technologies*

# Links

GitHub: *Link to repository*<sup>1</sup>

Notion: *Link to Notion team workspace*<sup>2</sup>

Demo Video: *Link to Drive*<sup>3</sup>

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Problem Definition</b>	<b>4</b>
<b>3</b>	<b>Related Work</b>	<b>5</b>
<b>4</b>	<b>System Architecture</b>	<b>6</b>
<b>5</b>	<b>Functional Requirements</b>	<b>7</b>
<b>6</b>	<b>System Design</b>	<b>8</b>
6.1	Database Schema . . . . .	8
6.2	Application design . . . . .	8
<b>7</b>	<b>Implementation</b>	<b>10</b>
7.1	Graphical User Interface . . . . .	10
7.1.1	Login and Registration . . . . .	11
7.1.2	Home Page . . . . .	11
7.1.3	Footer . . . . .	13
7.1.4	Editing user profile and following sections . . . . .	13
7.1.5	Home Page without Events . . . . .	14
7.1.6	Home Page for ESN Section . . . . .	14
7.1.7	Event Registration for Student . . . . .	14

---

<sup>1</sup><https://github.com/milivojcevic6/ticketing-app.git>

<sup>2</sup><https://shorturl.at/tAKM5>

<sup>3</sup>[https://drive.google.com/file/d/1RDunMQZZ7B8\\_xjlj00o00jSxPSTJxd-i/view?usp=sharing](https://drive.google.com/file/d/1RDunMQZZ7B8_xjlj00o00jSxPSTJxd-i/view?usp=sharing)

7.1.8	User Wallet . . . . .	16
7.1.9	QR Code Reader . . . . .	16
<b>8</b>	<b>Future Work</b>	<b>18</b>
<b>9</b>	<b>Contributions</b>	<b>18</b>
	<b>Appendix</b>	<b>20</b>
<b>A</b>	<b>Task Management</b>	<b>20</b>

# 1 Introduction

The Erasmus Student Network (ESN) is a renowned student organization, dedicated to supporting international students during their study abroad. ESN sections, established within universities worldwide, play a pivotal role in organizing events, trips, and social activities, fostering a welcoming and inclusive environment for international students.

ESN volunteers tirelessly dedicate their time and effort to ensure the smooth operation of these events, allowing international students to connect with the local community, make friends, and broaden their horizons. However, as the scale and complexity of the events increase, it becomes crucial to leverage the technology and streamline the administrative tasks to enhance the effectiveness and efficiency of event management.

In this context, the need for applications tailored to the needs of ESN volunteers becomes evident. These applications can provide centralized platforms for event management, simplifying processes such as event posting, attendee registration, ticket generation, and ticket scanning. By automating these tasks, ESN volunteers can focus more on creating meaningful experiences for international students, rather than getting caught up in administrative burdens.

Additionally, recognizing the valuable work done by ESN volunteers, ESN Awards competitions highlight and celebrate outstanding achievements within the organization. The IT Award segment specifically acknowledges the contributions of volunteers in leveraging the technology to enhance the ESN experience.

In the following sections, this document presents a problem statement, highlighting the challenges faced by ESN volunteers and proposes a solution that aligns with ESN's vision. By creating an application that addresses these challenges, ESN sections can efficiently manage events, simplify registrations, generate tickets, and streamline the check-in process. This comprehensive solution not only facilitates the work of ESN volunteers, but also enhances the overall experience of international students participating in ESN events.

## 2 Problem Definition

In most ESN sections, the current event registration process involves several manual steps, leading to inefficiencies and time-consuming tasks. The process typically includes the following:

1. Student registration: Students interested in attending an event fill out an online form, providing their details and preferences.
2. Excel-based registrations: ESN sections receive an Excel file containing the registration data, which needs to be manually downloaded, sorted, and organized by section volunteers.
3. In-person payments: Students are required to make payments in person, either at the ESN section office hours or during designated additional payment sessions.
4. Manual attendance marking: ESN sections mark the attendance of paid students manually in the Excel sheet, which can be error-prone and time-consuming.
5. Manual attendance checking: During the event, ESN volunteers cross-reference the Excel sheet to verify the attendance of registered students, resulting in a cumbersome process.

The current manual event registration process is outdated and inefficient, relying on Excel sheets and manual data management. It presents challenges in terms of accuracy, data organization, and the time spent by ESN volunteers. Additionally, maintaining real-time updates and seamless communication between ESN sections and event attendees becomes challenging, as well as data gets lost.

To address these issues, a technological solution is needed to simplify event registration, enhance data accuracy, and improve overall efficiency in event management.

### 3 Related Work

Various event management applications and ticketing platforms exist in the market, but most of them fail to meet the specific requirements and personalization needed for ESN sections. These applications often come with limitations and paid features that do not align with the goals, operations and the budget of ESN.

One significant challenge faced by ESN sections is the large number of students they cater to, especially in big sections where capacity can reach thousands of students. Existing applications often struggle to handle such volumes of attendees efficiently and effectively. Scalability becomes a critical concern, as managing event registrations, payments, and attendance for such a significant number of participants needs to be purchased in the existing applications (Figure 1).

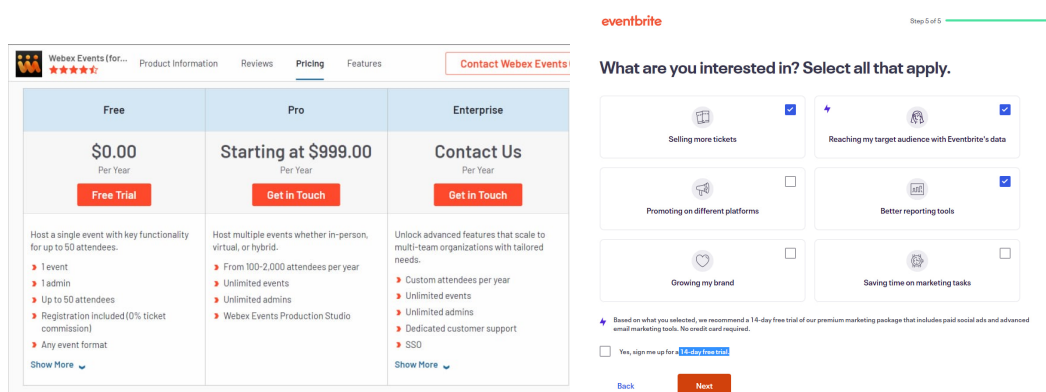


Figure 1: Free license events plans of existing event management applications offer very limited features.

Moreover, another crucial aspect for ESN is maintaining a consistent visual identity across its various platforms and applications. ESN has established visual identity guidelines that reflect its brand and values. However, existing applications often lack the ability to align with these visual identity rules, resulting in a disjointed and inconsistent user experience.

Recognizing these limitations, the development of the ESN Ticketing App aims to address these challenges directly. By focusing on the unique requirements of ESN sections and students, the app provides a personalized and scalable solution for event management. It ensures seamless handling of big student volumes, while also offering essential features like online payments and attendance tracking.

## 4 System Architecture

The proposed ESN Ticketing App follows the classical client-server architectural approach. The frontend is developed using React JS [3], a JavaScript library for building user interfaces, while the backend is built with Spring Boot [4], a Java framework for creating robust and scalable web applications. The frontend communicates with the backend through HTTP requests using the Axios library [2], a widely adopted JavaScript library for making asynchronous HTTP requests. The backend communicates with the frontend through RESTful APIs, which provide endpoints for various functionalities such as event retrieval, registration submission, payment processing, and QR code generation. These APIs are implemented using Spring MVC, which maps incoming requests to appropriate controller methods.

To interact with the MySQL database, the backend utilizes the Spring Data JPA, which is a Spring Boot module that provides an abstraction layer for database operations. This simplifies data access by allowing developers to work with Java objects rather than writing SQL queries directly. Spring Data JPA handles the mapping of objects to database tables, query generation, and transaction management.

Overall, the ESN Ticketing App follows a client-server architecture, where the React frontend communicates with the Spring Boot backend through HTTP requests (Figure 2). The backend processes these requests, interacts with the MySQL database using Spring Data JPA, and returns the appropriate responses to the frontend. This architecture ensures separation of concerns, scalability, and maintainability of the application.

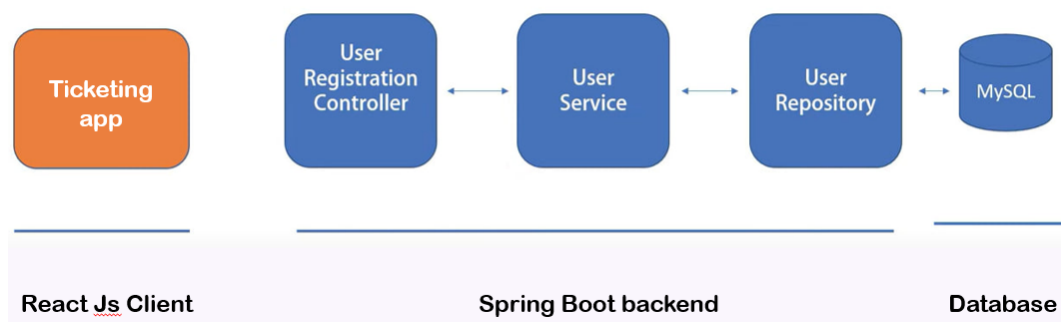


Figure 2: ESN Ticketing App system architecture

## 5 Functional Requirements

1. **Authentication and Authorization:** The app should provide secure authentication and authorization mechanisms to ensure that only authorized users can access and perform specific actions within the system. Users will need to authenticate using their registered email address and password to access the app. Role-based access control will be implemented to assign different levels of access and permissions based on user roles.
2. **Event Creation:** ESN sections should be able to create and publish events on the platform. They should be able to provide event details such as title, description, date, time, location, capacity, and any associated fees.
3. **Event Registration:** Students should be able to browse and search for events on the app. They should be able to register for events they are interested in attending by providing their personal details.
4. **Event Management:** ESN sections should have access to a dashboard or administrative interface where they can manage events. This includes editing event details, tracking event registrations, and viewing attendee lists.
5. **Ticket Generation:** After successful event registration, the app should generate a unique QR code ticket for each registered student, which can be stored and accessed from their profile.
6. **Ticket Scanning:** ESN volunteers should be able to use the app to scan QR codes on tickets to verify attendee presence at the event. The app should provide instant confirmation of the ticket's validity.
7. **User Profiles:** The app should allow both, students and sections, to view and manage their profiles and update information. Students should be able to add ESN sections to their list of followed sections, view registered events, and access their QR code tickets.



## 6 System Design

### 6.1 Database Schema

In the database schema of the ESN Ticketing App, the following entities and their relationships are defined (Figure 3):

1. **Section:** Represents an ESN section. It has a one-to-many relationship with the Event entity, as one section can organize multiple events, but those events can only be organized by that section.
2. **Event:** Represents an event, organized by a section. It has a one-to-many relationship with the Ticket entity, as each event can have multiple unique tickets associated with it.
3. **Ticket:** Represents a unique ticket for an event. It belongs to a specific event and has a one-to-many relationship with the User entity, as each ticket is associated with a single user who registered for the event.
4. **User:** Represents a student that uses the ESN Ticketing App. It has a one-to-many relationship with the Ticket entity, as a user can register for multiple events and have multiple tickets associated with them.
5. **UserSection:** Represents the relationship between a user and a section. It establishes a many-to-many relationship between the User and Section entities, as a section can have multiple users as members, and a user can follow multiple sections.

### 6.2 Application design

The ESN Ticketing App was designed from scratch, starting with a paper prototype to conceptualize its features and user interface. The frontend design of the app incorporates the official ESN colors and follows the ESN Visual Identity guidelines to maintain brand consistency. The app's logo was created using Canva [1], ensuring a visually appealing and recognizable brand identity. Bootstrap framework was utilized to enhance the frontend presentation, providing a responsive and visually appealing layout. On the backend, the app leverages the Spring Boot MVC framework, enabling efficient handling of requests and responses. The repositories, models, services, and controllers were developed from scratch, tailored to the specific requirements of the ESN Ticketing App. This custom backend design ensures a robust and scalable architecture to support the app's functionalities.

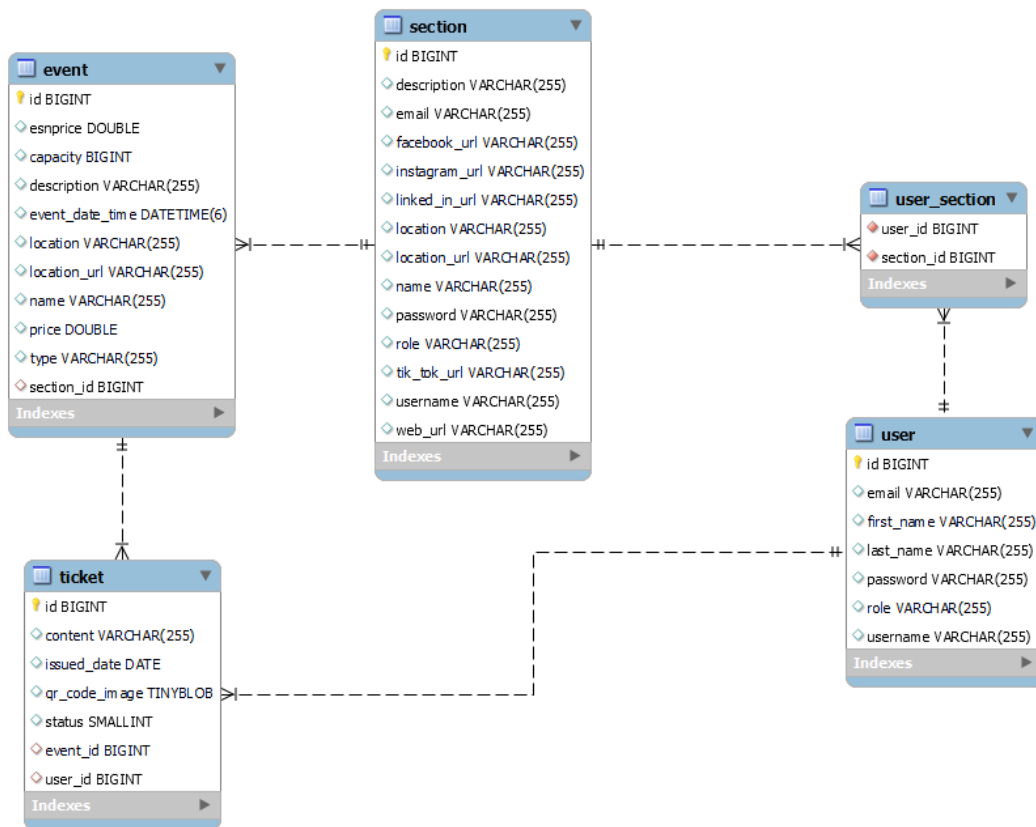


Figure 3: ESN Ticketing App database schema

## 7 Implementation

The implementation of the ESN Ticketing App involved the use of several technologies and tools. The development environment consisted of IntelliJ IDEA and JetBrains Rider as the primary integrated development environments (IDEs) for frontend and backend development, respectively. Postman was utilized for API testing, and MySQL Workbench was used for managing the database.

The codebase was structured into separate frontend and backend directories. Git, a distributed version control system, was employed for source code management and collaboration among team members.

In the backend, we implemented CRUD (Create, Read, Update, Delete) operations for the user, section, event, and ticket entities. This allowed for efficient management and manipulation of data related to these entities. Authentication and authorization were implemented using Spring Security, which provided robust security features and role-based access control. The backend consisted of controllers, models, repositories, and services for handling various functionalities related to authentication, events, sections, tickets, and users. Additionally, the `WebSecurityConfig` class was created to configure Spring Security and manage path access restrictions.

The frontend of the app was developed using React, a popular JavaScript library for building user interfaces. Axios, a promise-based HTTP client, was utilized to make API requests from the frontend to the backend. The frontend codebase included six components: Home, Login, Profile, TicketWallet, CheckTickets, and Footer. The `LoginContext` was used to store the current user information, while the `TicketWallet` component is accessible only to students, and the `CheckTickets` component is accessible only to the sections. A friendly phone view was also developed with a significant emphasis on the responsiveness of the app.

One of the significant challenges encountered during the implementation phase was integrating the latest version of Spring Security, which introduced significant changes compared to the previous version. Overcoming this challenge required thorough understanding and adaptation of the new features and configurations provided by Spring Security.

### 7.1 Graphical User Interface

The Graphical User Interface (GUI) of the ESN Ticketing App was meticulously designed to provide an intuitive and visually appealing experience for both, ESN sections and students. By incorporating the official ESN colors and following the ESN Visual Identity guidelines, the GUI maintains brand consis-

tency and fosters a sense of familiarity for users. The frontend development was implemented using React, a popular JavaScript library for building user interfaces, and the Bootstrap framework was utilized to ensure a responsive and aesthetically pleasing layout.

### 7.1.1 Login and Registration

The login and registration pages, represented in Figures 4 and 5, respectively, demonstrate a minimalist and user-friendly design. The login page provides a straightforward interface for existing users to access their accounts. If users are not logged in, they are automatically redirected to the login page. To register, users can easily switch to the registration form by selecting "Register" at the top of the form. The registration page (Figure 5) adapts the login form to include the necessary fields for user registration.



Figure 4: Login Page

### 7.1.2 Home Page

Figure 12 showcases the home page of the app, which serves as the primary entry point for users. The design offers an engaging layout, with options for students to explore and search for events they are interested in attending. The responsive design of the home page ensures a seamless user experience for both desktop and mobile users (Figure 7), allowing them to conveniently access event information and registration options.



Figure 5: Registration Page

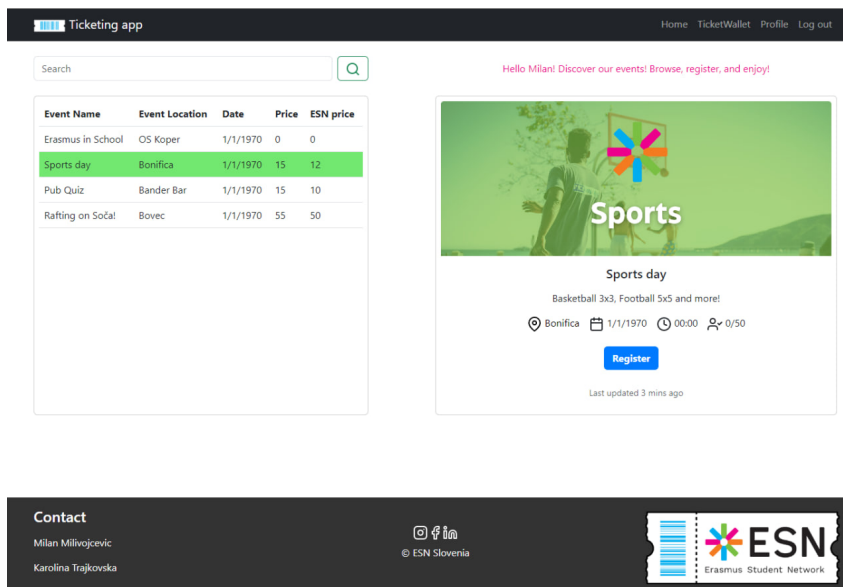


Figure 6: Home Page

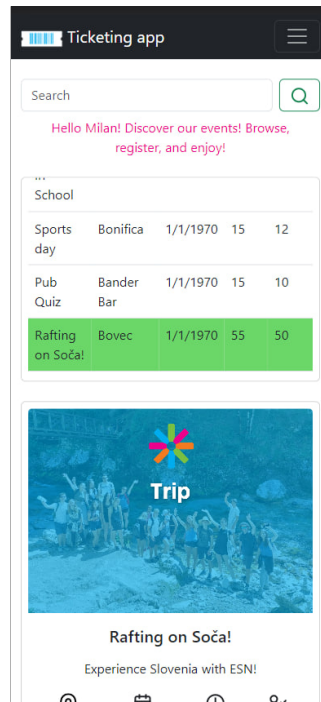


Figure 7: Home Page - Mobile View

### 7.1.3 Footer

The footer section of the app (Figure 8) displays essential information, including the names of the developers, the ESN organization, and the app logo. It also offers easy access to ESN's social media links, promoting smooth communication and engagement with the ESN community. The inclusion of the app logo enhances brand recognition, insuring that users can easily identify and relate to the app within the context of the ESN ecosystem.

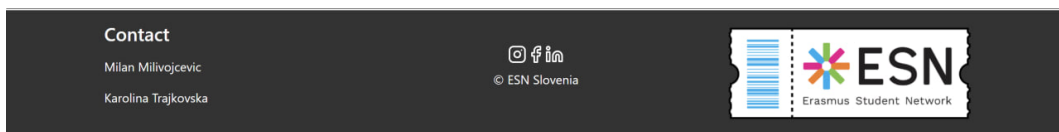


Figure 8: Footer

### 7.1.4 Editing user profile and following sections

The user profile page, as shown in Figure 9, offers users the option to edit certain attributes, including their first name, last name, and password. Ad-

ditionally, users can select ESN sections to follow, facilitating personalized event recommendations and a tailored experience within the app.

The screenshot shows the 'User Profile' editing interface. At the top, there is a navigation bar with 'Ticketing app' on the left and 'Home TicketWallet Profile Log out' on the right. The main heading is 'User Profile'. Below it, there are several input fields: 'Username' with the value 'Milan', 'Email' with 'milan@gmail.com', 'First Name' with 'Milan', and 'Last Name' with 'e.g. Milivojevic'. There is also a 'New Password' field with the placeholder text 'Change your password'. To the right of the password field is a 'Sections' dropdown menu that is open, showing three options: 'Please choose an option...', 'zifmorska', and 'maribor'. Below the form is a blue 'Save Changes' button. At the bottom of the screen, there is a footer with contact information: 'Contact Milan Milivojevic Karolina Trajkovska', social media icons for Instagram, Facebook, and LinkedIn, and the ESN logo with the text 'Erasmus Student Network'.

Figure 9: Editing user profile and choosing sections to follow

### 7.1.5 Home Page without Events

If there are no events for a specific section, the home page of that section displays a clean and informative layout, as seen in Figure 10. This design ensures that users, especially ESN sections, are aware of the current status and allows them to focus on creating and managing events. The GUI effectively communicates the absence of events, streamlining the event management process for ESN sections.

### 7.1.6 Home Page for ESN Section

Figure 11 illustrates the home page view, tailored specifically for ESN sections. This view provides a comprehensive overview of all events organized by the section, allowing section administrators to easily track and manage their events. The home page design enables efficient event editing and ensures a smooth user experience for section administrators.

### 7.1.7 Event Registration for Student

Once a student user has specified the sections they wish to follow, the home page view (Figure 12) displays events from the followed sections. This en-

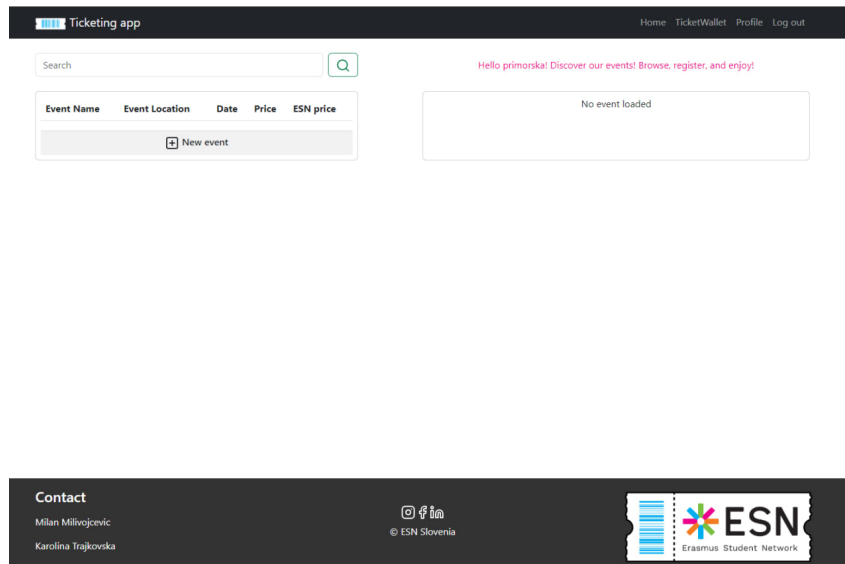


Figure 10: Home Page of a section

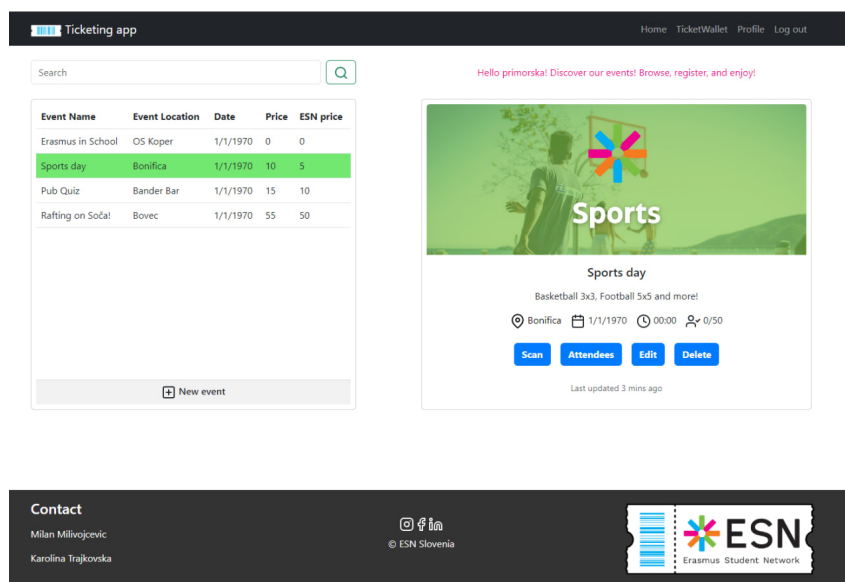


Figure 11: The Home Page of the ESN Section



ables students to conveniently explore and register for events that align with their interests, streamlining the event registration process.

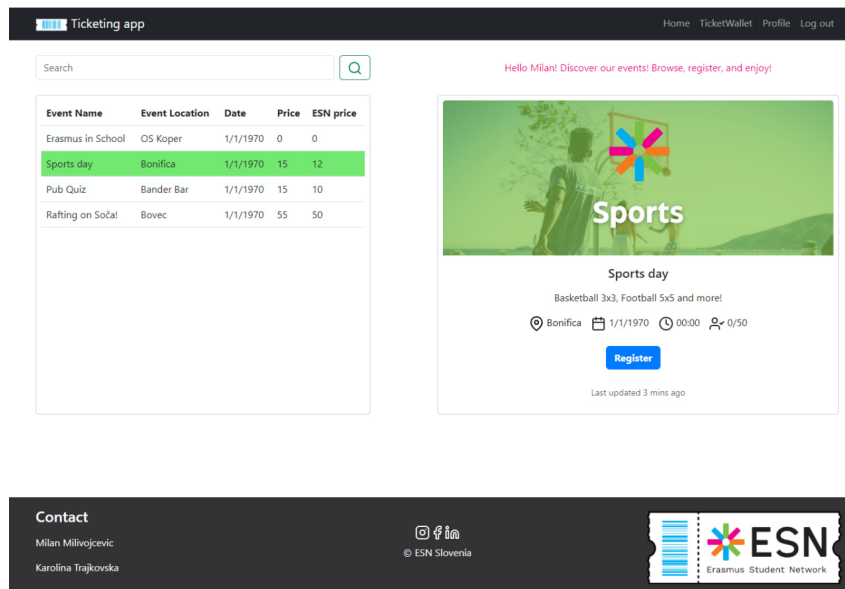


Figure 12: Event Registration for Student

### 7.1.8 User Wallet

Figure 13 shows the page design for the user's wallet, where students can access their QR code tickets for registered events. The user wallet provides a comprehensive view of their registered events, and offers easy navigation for mobile users.

### 7.1.9 QR Code Reader

Figure 14 illustrates the QR code reader functionality, displaying the information contained in the QR code. The QR code provides essential details, including the event name, the user's full name, and the date when the ticket was created. This feature enables seamless and efficient ticket scanning during event check-ins.

The thoughtfully crafted User Interface of the ESN Ticketing App ensures an enjoyable and efficient user experience. It reflects the dedication of the development team to creating an application that aligns with the needs and visual identity of ESN. The UI not only streamlines event management processes for ESN sections but also offers a seamless event registration and

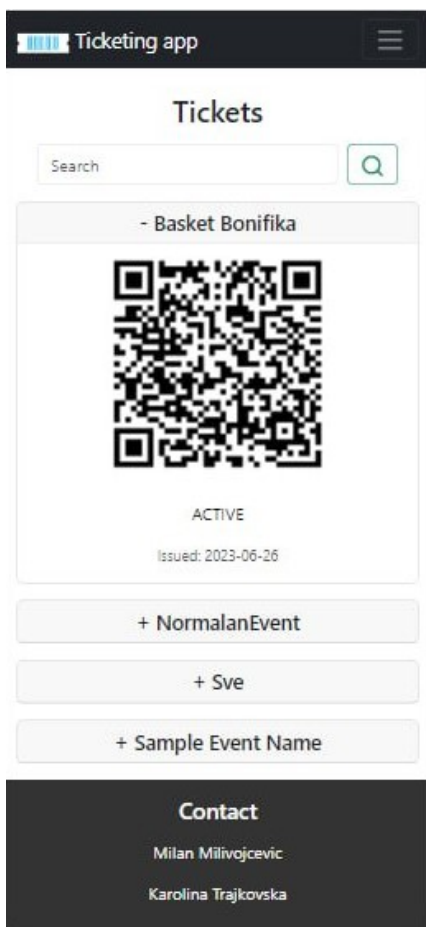


Figure 13: Ticket Wallet

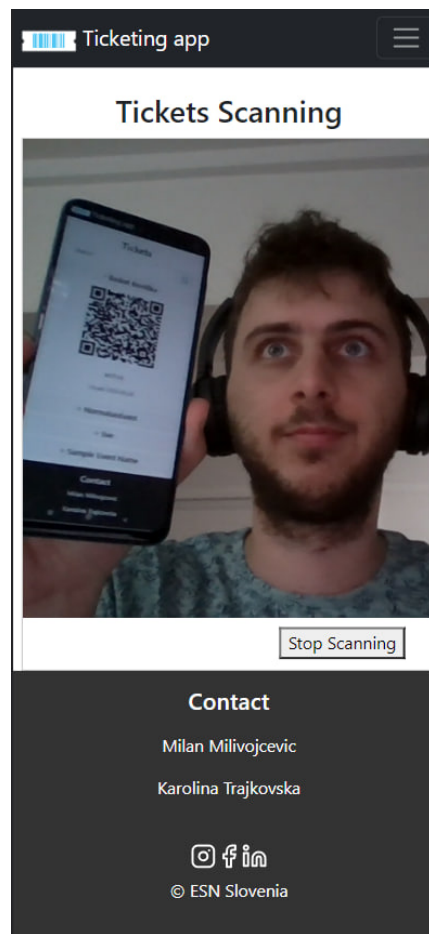


Figure 14: QR Code Reader

ticketing journey for students, ultimately enhancing the overall effectiveness and engagement of ESN events.

## **8 Future Work**

In the future, we have identified several areas for improvement and additional features to enhance the ESN Ticketing App. Our next steps will involve improving the user experience by implementing a more intuitive and visually appealing interface. We aim to focus on refining the user interface design and optimizing the app's usability to provide a seamless event registration and ticketing experience for both ESN sections and students.

Moreover, we plan to integrate a payment system and deploy the app for testing during the upcoming semester, specifically during orientation days events. Evaluating the app's performance in a high-traffic scenario will provide valuable insights.

Additionally, we aim to incorporate event statistics to help ESN sections make informed decisions. On the user side, implementing a recommender system will enhance the app by providing personalized event suggestions. These future steps will ensure a more user-friendly, efficient, and engaging experience for both ESN sections and students.

## **9 Contributions**

Throughout the development of the ESN Ticketing App, the project was carried out by a developer team consisting of two members: Karolina Trajkovska and Milan Milivojčević. Initially, both team members were involved in full-stack development, working collaboratively to tackle various aspects of the project.

To ensure efficient task management, a Notion task tracking system was implemented, allowing the team to organize and assign specific tasks from specific sprints (agile development). During this phase, Karolina and Milan worked closely together, dividing the tasks based on their expertise and workload (Appendix A).

As the development progressed, a natural division of tasks occurred, with Karolina primarily focusing on frontend development and Milan taking charge of backend tasks. This division of responsibilities allowed for increased productivity and specialization in their respective areas of expertise.

Despite the specialization, the team maintained a collaborative approach throughout the project. They continuously supported and helped each other in debugging and problem-solving, fostering a cohesive and dynamic develop-

ment environment. Both team members have a comprehensive understanding of the overall project structure and functionalities.

Throughout the development process, the team received valuable guidance and support from the mentor, Dr. Domen Šoberl. Dr. Šoberl's expertise and insights in the field of software development were instrumental in shaping the direction of the project. He provided the team with valuable advice on architectural design, best practices, and helped them navigate challenges encountered along the way.

## References

- [1] *Free design tool: Presentations, video, social media | CANVA*. [Accessed 13-08-2023].
- [2] *Getting Started | Axios Docs – axios-http.com*. [Accessed 13-08-2023].
- [3] *React. React – react.dev*. Accessed 12-08-2023. URL: <https://react.dev/>.
- [4] *Spring Boot. Spring Boot – spring.io*. Accessed 12-08-2023. URL: <https://spring.io/projects/spring-boot>.

# Appendices

## A Task Management

**Sprints** Beta

Timeline All + Filter Sort Q ...

Aa Sprint name	Σ Is Current Sp...	March 2023	April	May	June	July	Year
		13 20 27	3 10 17 24	1 8 15 22 29	5 12 19 26 3	10 17 24 31	
🏠 Sprint 1	<input type="checkbox"/>	🏠 Sprint 1 March 13, 2023 → March 26, 2023					
🏠 Sprint 2	<input type="checkbox"/>		🏠 Sprint 2 March 27, 2023 → April 24, 2023				
🏠 Sprint 3	<input type="checkbox"/>			🏠 Sprint 3 April 24, 2023 → June 17, 2023			
🏠 Sprint 4	<input type="checkbox"/>				🏠 Sprint 4 June 17, 2023 → June 30, 2023		

### Tasks

By project This sprint Backlog Mine People + Filter Sort Q ... New

Project Status: To-do, In Progress, ... Assign Due Sprint Project + Add filter

1. Forming Team 3 ... +

Aa Task name	Status	Assign	Due	Sprint	Project
📄 Look for other members	Done	K Karolina Trajkovska M Milan Milivojce	March 8, 2023	🏠 Sprint 1	1. Forming Team
📄 Discuss topics	Done	K Karolina Trajkovska M Milan Milivojce	March 10, 2023	🏠 Sprint 1	1. Forming Team
📄 Approach a mentor	Done	K Karolina Trajkovska M Milan Milivojce	March 13, 2023	🏠 Sprint 1	1. Forming Team

+ New COMPLETE 3/3

2. Researching the market 2 ... +

Aa Task name	Status	Assign	Due	Sprint	Project
📄 Find Similar Applications	Done	K Karolina Trajkovska M Milan Milivojce	March 15, 2023	🏠 Sprint 1	2. Researching the n
📄 Check functionalities and limitation	Done	K Karolina Trajkovska M Milan Milivojce	March 20, 2023	🏠 Sprint 1	2. Researching the n

▼ 3. Planning 5 ... +

Aa Task name	Status	Assign	Due	Sprint	Project
Decide on technology	Done	Karolina Trajkovska, Milan Milivojevic	April 2, 2023	Sprint 2	3. Planning
Infrastructure	Done	Milan Milivojevic, Karolina Trajkovska	April 2, 2023	Sprint 2	3. Planning
Database design	Done	Milan Milivojevic	April 4, 2023	Sprint 2	3. Planning
User types	Done	Karolina Trajkovska	April 4, 2023	Sprint 2	3. Planning
User Interface (UI)	Done	Karolina Trajkovska	April 8, 2023	Sprint 2	3. Planning

▼ 4. Developing 13 ... +

Aa Task name	Status	Assign	Due	Sprint	Project
Create Database	Done	Karolina Trajkovska	May 31, 2023	Sprint 3	4. Developing
Setup environment	Done	Milan Milivojevic	May 31, 2023	Sprint 3	4. Developing
Navbar & Footer	Done	Milan Milivojevic	May 31, 2023	Sprint 3	4. Developing
Events page	Done	Karolina Trajkovska	May 31, 2023	Sprint 3	4. Developing
Ticket Wallet page	Done	Karolina Trajkovska	May 31, 2023	Sprint 3	4. Developing
Profile page	Done	Milan Milivojevic	May 31, 2023	Sprint 3	4. Developing
Scan Ticket page	Done	Milan Milivojevic	May 31, 2023	Sprint 3	4. Developing
CRUD User	Done	Milan Milivojevic	May 31, 2023	Sprint 3	4. Developing
CRUD Event	Done	Karolina Trajkovska	May 31, 2023	Sprint 3	4. Developing
CRUD Ticket	Done	Milan Milivojevic	May 31, 2023	Sprint 3	4. Developing
Axios API (fetch data)	Done	Karolina Trajkovska	May 31, 2023	Sprint 3	4. Developing
Payment System	Not Started	Milan Milivojevic, Karolina Trajkovska			4. Developing

▼ 5. Testing 2 ... +

Aa Task name	Status	Assign	Due	Sprint	Project
Alpha test (one user tester)	Done	Karolina Trajkovska, Milan Milivojevic	June 10, 2023	Sprint 4	5. Testing
Section testing	Not Started	Karolina Trajkovska, Milan Milivojevic	June 30, 2023	Sprint 4	5. Testing

+ New

COMPLETE 1/2

▼ 6. Finalizing 3 ... +

Aa Task name	Status	Assign	Due	Sprint	Project
Evaluation	In Progress	Karolina Trajkovska, Milan Milivojevic	June 15, 2023	Sprint 4	6. Finalizing
Writing Documentation	Done	Karolina Trajkovska	June 15, 2023	Sprint 4	6. Finalizing
Final presentation	Done	Milan Milivojevic	June 15, 2023	Sprint 4	6. Finalizing