Runexus
Requirements Document

Irmak Damla Özdemir, 5123127
irmakdamla.oezdemir@study.thws.de

Buse Okcu, 5123129
buse.okcu@study.thws.de

1 Project Overview

Title: Runexus – Running Event Booking and Runner Buddy Matching Platform

Runexus is a platform designed to simplify the discovery, organization, and participation in local running events that enables runners to connect with others who share similar paces, encouraging community building and shared experiences.

Target Users:

- Individual runners across all levels, from recreational to community-based participants
- Running groups and event organizers
- Sports enthusiasts interested in joining running activities

Application Goals:

- Simplify the process of organizing and discovering local running events
- Connect and match runners with similar paces
- Support the growth of a connected running community

2 Key Features

2.1 Must-Have

- **User Authentication:** Users can register, log in, and log out securely.
- **Forum Posting:** Users can create and publish forum posts to interact with the community.
- **Event Management:** Users can create new events, update event details, and delete their own events.
- **Event Participation:** Events include a ”Join Event” button and dynamically display the number of remaining spots (e.g., ”3 spots left”).
- **Event Commenting:** Users can leave comments on different events to express interest or ask questions.

2.2 Nice-To-Have

- **Account Deletion:** Users can permanently delete their accounts along with all associated data.
- **User Following:** Users can follow other runners to stay updated on their activities and upcoming events.
- **Direct Messaging (DM):** Matched or followed users can exchange private messages through an integrated chat feature.
- **Pace-Based Matching:** Users can be matched with others based on their average running pace.
- **Match Connections:** Matched users are displayed in a dedicated list and can be accessed via a ”Match” button.

3 User Roles and Interactions

- **User:** A registered and logged-in user with full access to all features of the platform. Users can create or join events, interact with others, and personalize their experience.
  - **Actions:**
    - Register, log in, and log out
    - Create, edit, and delete their own events
    - Join events and see available spots
    - Comment on events
    - Set pace and match with other users
    - View matched users and send direct messages
    - Follow other users
    - Create forum posts and participate in discussions
    - Delete their own account
- **Visitor:** A non-registered user who can explore the platform without an account. Visitors have limited access and can only view publicly available content.
  - **Actions:**
    - View homepage
    - browse event listings
    - Join events and see available spots
    - read forum posts (read-only)

4 User Stories / Use Cases

1. A visitor can view the running events, available spots and comments.

2. A new user can register and log in so that the user can fully access the platform and join running events.

3. A registered user can create a new running event by selecting a time, date, and location.

4. A registered user can specify her/his own average running pace and be matched with others who run at a similar speed.

5. A matched user can send private messages and follow the matches so they can stay connected and plan runs together.

6. A registered user can view the number of available spots in an event and quickly reserve a place before the event reaches full capacity.

7. A registered user can write and read comments on event pages to ask questions, coordinate with others.

5 Non-Functional Requirements

- **Usability:** The application should provide an intuitive and user-friendly interface that follows common interaction patterns, minimizing the functional learning curve for new users.
- **Responsiveness:** The user interface must adapt seamlessly to various screen sizes, ensuring a consistent experience on both desktop and mobile devices.
- **Accessibility:** All interactive elements should be properly labeled and structured to be compatible with screen readers and assistive technologies, promoting inclusive access.
- **Performance:** Application pages should load within 2 seconds under standard network conditions to ensure a smooth and responsive user experience.

6 Technology Assumptions

- **React:** Used for building the user interface with reusable components.

- **CSS Modules or plain .css files:** For styling the application, depending on the component structure and complexity.

- **REST API:** To communicate with the backend.

- **Docker with a Web Server:** For containerizing and serving the frontend application in a production-ready environment.

7 Project Constraints

- Deadline: July 30, 2025

- All functionality must be built in React and dockerized.

- DM may be mocked or partially implemented.

- Backend integration depends on available endpoints or mock APIs.

8 Acknowledgment of AI Assistance

This document was drafted and refined using GPT-4 based on our team's outline and requirements. The team reviewed, edited, and structured the generated content to ensure quality, consistency, and relevance to the project and assignment goals.