

# Unofficial Web Development Competition 2025

---

## TEST PROJECT

Written by

Margit Tennosaar  
Franz Stimpfl

## Table of Contents

<b>Introduction .....</b>	<b>3</b>
Technical requirements .....	3
<b>Welcome page .....</b>	<b>4</b>
User Story 1: Add SVG Animation .....	4
User Story 2: Include a CTA .....	4
<b>Setup page .....</b>	<b>5</b>
User Story 1: User Information Input .....	5
User Story 2: Join Existing Community .....	5
User Story 3: Create a New Community .....	5
<b>Community page .....</b>	<b>6</b>
User Story 1: Offer Home Screen Shortcut .....	6
User Story 2: Remember Returning Users .....	6
User Story 3: View Current Exercise .....	6
User Story 4: Mark the Exercise as Completed .....	7
User Story 5: Track Participation .....	7
<b>Admin page .....</b>	<b>8</b>
User Story 1: Manage Exercises .....	8
User Story 2: Restrict Access .....	8
<b>Owner page .....</b>	<b>9</b>
User Story 1: View All Communities .....	9
User Story 2: Community Settings .....	9
User Story 3: Remove Communities .....	9
<b>UI &amp; Creativity .....</b>	<b>10</b>
User Story 1: Brand Identity .....	10
User Story 2: Consistent UI Design .....	10
User Story 3: Delightful Details .....	10
<b>Code Quality &amp; Delivery .....</b>	<b>11</b>
User Story 1: Refactor Outdated Code .....	11
User Story 2: Maintainable Code .....	11
User Story 3: Performance Matters .....	11
User Story 4: Deliver Properly .....	11
<b>Marking Scheme .....</b>	<b>13</b>

# Introduction

## ActivePulse – Your Daily Dose of Fun Fitness!

Welcome to ActivePulse, a community-driven app that makes staying active fun and social. At random times each day, members of your community receive a challenge—something simple like push-ups or squats. Your goal? Do the exercise and log it as fast as you can. Your reaction speed is tracked, consistency is celebrated, and the results help build motivation.

The app updates in real-time as people join and participate. You'll always know who completed the latest challenge and how fast they were. Each community has its unique URL, making inviting friends or building a team easy.

It's not about intense workouts—it's about showing up. A few minutes a day to move, compete, and connect.

App slogan: **"Stay Ready. Stay Active. Stay Together."**

## Technical requirements

- Build as a Progressive Web App (PWA)
- Use WebSockets for real-time updates
- Design with a mobile-first approach
- Use any modern web framework or library
- Deploy on provided infrastructure
- The app must run through the provided URLs
- The project must be version-controlled with Git
- Keep AI use to a minimum – this is competition preparation

Competition time: 7 hours

## Welcome page

The welcome page is the first page users see. It's simple and welcoming, introducing the app's concept and inviting users to start.

URL: /

### User Story 1: Add SVG Animation

*As a user, I want to see a lightweight, looped SVG animation so that the page feels dynamic and fun.*

Acceptance Criteria:

- An SVG animation is displayed in the layout
- The animation runs smoothly in looped mode
- Animation is optimised and does not have a long loading time

### User Story 2: Include a CTA

*As a user, I want a call-to-action button so that I know how to begin using the app.*

Acceptance Criteria:

- The button text clearly invites the user to start
- The button links to the Setup Page

## Setup page

The setup page is where users join the app for the first time. They can either join an existing community or create a new one. Everyone fills in their name and selects an avatar from the ones we provide. Once they enter, their name and avatar will appear on the Community Page.

URL: /setup

### User Story 1: User Information Input

*As a user, I want to enter my name and choose an avatar so that my profile is set up.*

Acceptance Criteria:

- The user can enter their name
- The user can choose an avatar from the provided options
- The entered name is stored
- The selected avatar is stored

### User Story 2: Join Existing Community

*As a user, I want to join an existing community by typing in a name so that I can participate.*

Acceptance Criteria:

- Input field available for joining by community slug
- If the community does not exist, the user is shown an error message
- If a community exists, a user is redirected to the Community Page

### User Story 3: Create a New Community

*As a user, I want to create a new community so that I can invite others and manage the group.*

Acceptance Criteria:

- The user can create a community with a custom name (slug)
- A unique admin URL is generated and displayed
- 10 exercises are added by default
- A first exercise is automatically started

## Community page

The community page is the heart of the app. Users can view the current exercise, respond to it, see who else participated, and check community stats. The page updates in real-time using WebSockets.

URL format: /[community-slug]

### User Story 1: Offer Home Screen Shortcut

*As a mobile user, I want to be able to save the app to my home screen so that I can access it like a native app.*

Acceptance Criteria:

- A "Save to Home Screen" button or prompt is available on mobile
- Button triggers browser-native PWA install flow
- The user can choose to add the app icon to their device's home screen

### User Story 2: Remember Returning Users

*As a returning user, I want my name and avatar to be remembered so that I don't have to enter them daily.*

Acceptance Criteria:

- Name and avatar are saved in the localStorage on the user's device
- When a user revisits the Community Page, the saved data is used automatically
- If data is not found, user is redirected to Setup Page

### User Story 3: View Current Exercise

*As a community member, I want to see the current exercise and how long ago it was posted.*

Acceptance Criteria:

- The current exercise is shown on the page
- Timer shows how much time has passed since it was posted
- Exercise updates in real-time when a new one is issued

## User Story 4: Mark the Exercise as Completed

*As a user, I want to mark the current exercise as completed so that my participation is counted.*

Acceptance Criteria:

- A "Mark as Done" button is visible when an exercise is active
- An optional input allows users to enter the number of reps they completed
- When clicked, the user's name, avatar, reps, and timestamp are recorded
- Reaction time is calculated from exercise start time
- The button is disabled or hidden after submission
- The user sees a confirmation message after submitting
- The completed entry appears in the real-time participant list

## User Story 5: Track Participation

*As a user, I want to see who completed the exercise and how fast they were.*

Acceptance Criteria:

- The list of participants is shown, with avatars and names
- Reaction time and reps completed are displayed
- List updates in real-time without refreshing

## Admin page

Each community has a unique admin page accessible via a secret slug. This page allows admins to manage their community's exercises.

URL format: /[community]/[admin-slug]

### User Story 1: Manage Exercises

*As an admin, I want to add, remove, or edit exercises so that I can keep things fresh.*

Acceptance Criteria:

- Admin can see a list of existing exercises
- Admin can manage exercises from a fixed list of checkboxes
- Admin can select or deselect exercises from the provided list
- Admin can assign a new exercise from the selected pool

### User Story 2: Restrict Access

*As an admin, I want the admin page to be private so I can only manage my community.*

Acceptance Criteria:

- Admin page is only accessible using a secret slug



## Owner page

This is the app's top-level control panel. As the app owner, you'll use this to see and manage all active communities.

URL: /5yo8sj

### User Story 1: View All Communities

As the owner, I want to see every community that exists so that I can monitor activity.

Acceptance Criteria:

- A list of all the communities is shown
- For each community, show the number of participants and number of exercises

### User Story 2: Community Settings

As the owner, I want to set size or participation limits for the community if needed.

Acceptance Criteria:

- Admin can adjust participant limits for each community
- Admin can adjust exercise count limits for each community

### User Story 3: Remove Communities

As the owner, I want to delete communities so that I can maintain quality.

Acceptance Criteria:

- Admin can remove a community
- Confirmation required before deletion
- Changes are reflected immediately in the community list

## UI & Creativity

This app should feel polished, modern, and joyful to use. You are encouraged to add thoughtful design touches that enhance usability and match its energy. Keep it simple and user-friendly.

### User Story 1: Brand Identity

As a UI developer, I want to apply the brand colors and logo so that the app feels unified and recognisable.

Acceptance Criteria:

- Brand colors are used consistently across all views
- The app logo is shown on the Welcome Page and header to all pages
- Visual style supports the app's theme and tone

### User Story 2: Consistent UI Design

As a user, I want the entire interface to feel clean and consistent so that my experience is smooth and intuitive.

Acceptance Criteria:

- Typography, layout, and components follow the same design rules
- Mobile layout is thoughtfully adapted (not just scaled)
- Visual hierarchy helps guide user actions

### User Story 3: Delightful Details

As a user, I enjoy small UI touches and animations that make the app feel lively.

Acceptance Criteria:

- Small animations or UI transitions enhance user experience
- Transitions are fast, subtle, and responsive

# Code Quality & Delivery

Your code should be clean, organised, and optimised. You will be given an outdated jQuery version of this page. Your first task is to refactor it completely using modern best practices. This includes everything from performance to how well other developers can understand and maintain your work.

## User Story 1: Refactor Outdated Code

*As a developer, I want to refactor the old jQuery welcome page so that it follows modern, maintainable standards.*

Acceptance Criteria:

- jQuery is entirely removed from the page
- Modern JavaScript is used
- The code is well-structured and clean

## User Story 2: Maintainable Code

*As a developer, I want to write structured, modern code so that the app is stable and easy to expand.*

Acceptance Criteria:

- Clear folder and component structure
- Meaningful naming conventions are used throughout
- Reusable functions and components are applied where appropriate

## User Story 3: Performance Matters

*As a user, I want the app to load and run quickly so that I stay engaged.*

Acceptance Criteria:

- No unnecessary re-renders or blocking operations
- Assets and data are loaded efficiently
- Page speed is optimised (target: Lighthouse performance score 91+)

## User Story 4: Deliver Properly

*As a judge, I want to review a professional-grade submission that reflects competition readiness.*

Acceptance Criteria:

- A complete and well-written README file is provided
- Project runs out of the box on provided infrastructure ([competitor-slug]-[module-slug].uwdc.skill17.com)
- All URLs work as expected (Welcome, Setup, Community, Admin, Owner)
- The primary entry point for the project should be the frontend module

## Marking Scheme

A	Welcome page	6.00
B	Setup page	17.00
C	Community page	29.00
D	Admin page	8.00
E	Owner page	11.00
F	UI & creativity	14.00
G	Code Quality & Delivery	15.00
Total		100.00