Software Requirements Document

Krav Maga Shadow Fighting Trainer PWA

Version 1.0
Date: August 22, 2025
Author: Grok, xAI

Contents

1 Introduction

1.1 Purpose

This Software Requirements Document (SRD) outlines the requirements for a Progressive Web App (PWA) designed to enhance shadow fighting training for Krav Maga practitioners. The app addresses the limitation in traditional shadow fighting where practitioners anticipate opponent moves, by randomly announcing techniques via audio, allowing realistic reaction training. The app uses WAV audio files, supports user-configurable session parameters, and is mobile-first with no installation required.

1.2 Scope

The app includes:

- Storage and playback of audio files for Krav Maga techniques.
- User configuration for fight duration, delay between actions, and technique prioritization.
- Random technique announcement during sessions.
- Mobile-responsive PWA design.

Out of scope: user authentication, hardware integration, analytics, audio file editing.

1.3 Definitions and Acronyms

- **PWA**: Progressive Web App A web application installable on devices, works offline.
- **Technique**: A Krav Maga action or attack, linked to a WAV file (e.g., "Straight Punch.wav").
- **Fight Duration**: Total training session time.
- **Delay Between Actions**: Time after an announcement for reaction.
- **Prioritization**: User-defined weights for technique selection probability.

1.4 References

- Krav Maga techniques: Black Belt Wiki, Wikipedia.
- PWA guidelines: MDN Web Docs (https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps).

2 Overall Description

2.1 Product Perspective

The app introduces unpredictability to shadow fighting, simulating real opponent actions via audio cues to improve reaction times and adaptability.

2.2 Product Functions

- Load and manage WAV audio files for techniques.
- Allow configuration of session parameters.
- Conduct sessions with random audio announcements.
- Support technique prioritization.

2.3 User Classes and Characteristics

- **Primary Users**: Krav Maga practitioners (beginner to advanced).
- **Assumptions**: Familiarity with smartphones/browsers, device with audio playback.

2.4 Operating Environment

- Browsers: Chrome, Safari, Firefox (mobile/desktop).
- Devices: Primarily mobile (iOS/Android), desktop supported.
- Offline capability: Cache assets for offline use.

2.5 Design and Implementation Constraints

- Use HTML5, CSS3, JavaScript.
- Service Workers for PWA features.
- Audio via Web Audio API.
- Client-side logic only.

2.6 Assumptions and Dependencies

- Pre-provided audio files bundled with app.
- Browser support for PWA and audio playback.

3 Functional Requirements

3.1 Technique Library Management

- Include predefined Krav Maga techniques, each linked to a WAV file named after the technique.
- Sample techniques:

- Load WAV files from assets directory, verify on initialization.

Category	Techniques
Punches	Straight Punch, Cross Body Punch, Hook Punch, Uppercut Punch
Strikes	Palm Heel Strike, Hammer Fist, Elbow Strike, Eye Strike, Knife Hand Strike
Kicks	Front Kick, Round Kick, Side Kick, Back Kick, Heel Kick, Spinning Outside Slap Kick
Knees	Vertical Knee Strike
Defenses/Grabs	Defense Against Front Choke, Defense Against Side Headlock, Defense Against Bear Hug, Escape From Mount
Weapons	Defense Against Knife Stab, Defense Against Gun Front

Table 1: Krav Maga Techniques

3.2 User Configuration

- Configure fight duration (1–30 minutes, 1-minute increments).
- Configure delay between actions (1–10 seconds, 0.5-second increments).
- Select and prioritize techniques (include/exclude, High/Medium/Low priority; e.g., High: 50%, Medium: 30%, Low: 20%).
- Save configurations locally (via LocalStorage).

3.3 Training Session

- Randomly select and play a techniques WAV audio, weighted by priorities.
- Wait for configured delay before next announcement.
- Continue until duration elapses or user stops.
- Display remaining session time.
- Optional: Show technique name for accessibility.

3.4 Session Controls

- Start, Pause, Stop buttons.
- Volume control.
- End-of-session summary (e.g., number of announcements, techniques used).

4 Non-Functional Requirements

4.1 Performance

- Audio playback latency < 100ms.
- App load time < 5 seconds on mobile networks.
- Support up to 100 techniques.

4.2 Usability

- Mobile-first, responsive UI (Bootstrap/Tailwind CSS).
- Intuitive single-page app.
- Accessibility: ARIA labels, keyboard navigation, high-contrast mode.

4.3 Reliability

- Handle audio errors gracefully.
- Offline functionality via Service Worker.

4.4 Security

- No user data collection; local operations.
- Secure asset loading (HTTPS).

4.5 Maintainability

- Modular code.
- Easy addition of new techniques.

4.6 Portability

- Cross-browser compatibility.
- iOS and Android support.

5 Supporting Information

5.1 Data Requirements

- Technique data: JSON array (name, file path, priority).

5.2 User Interface Sketches

- **Home Screen**: Configuration form (sliders for duration/delay, checklist for techniques with priority dropdowns).
- **Session Screen**: Timer, Start/Pause/Stop buttons, optional technique text.

5.3 Testing Considerations

- Unit tests for random selection.
- Integration tests for audio playback.
- Usability testing on mobile devices.