

Product Requirements Document (PRD): Wi-Fi Based Attendance System

Version: 1.0 **Date:** September 5, 2025 **Author:** Gemini **Status:** Draft

1. Introduction

This document outlines the product requirements for a web-based attendance system designed to run on a Raspberry Pi within the office network. The system will allow employees to mark their attendance only when they are connected to the designated office Wi-Fi network. This ensures that employees are physically present in the office when they mark themselves as "present." The application will be built using React.

2. Vision & Goals

2.1. Vision

To create a simple, reliable, and cost-effective attendance system that leverages existing office infrastructure (Wi-Fi and a Raspberry Pi) to ensure accurate presence verification.

2.2. Goals

- **Primary Goal:** Develop a system where employees can mark attendance only if they are connected to the office Wi-Fi.
- **Simplicity:** Create an intuitive and easy-to-use interface for both employees and administrators.
- **Accuracy:** Eliminate proxy attendance by linking it to the physical presence required for a Wi-Fi connection.
- **Low Cost:** Utilize a Raspberry Pi as a low-power, dedicated server, minimizing hardware costs.
- **Data Accessibility:** Provide administrators with a clear view of attendance records.

3. User Personas & Roles

3.1. Employee

- **Description:** A regular employee of the company.
- **Goals:** To quickly and easily mark their daily attendance. To view their own attendance history.
- **Frustrations:** Forgetting to sign in, complicated login processes, uncertainty if their attendance was recorded.

3.2. Administrator (e.g., HR Manager, Office Admin)

- **Description:** A user responsible for managing and viewing attendance records for all employees.
- **Goals:** To get an accurate overview of daily and historical attendance. To manage the employee list. To export attendance data for payroll or records.
- **Frustrations:** Inaccurate records, manual data entry, difficulty in generating reports.

4. Features & Functional Requirements

4.1. Core Feature: Wi-Fi Based Check-in

- **Requirement 4.1.1:** The system must be able to verify that the employee's device (laptop, phone) is connected to a specific office Wi-Fi network (based on SSID).
- **Requirement 4.1.2:** The "Mark Attendance" button shall be enabled *only* if the device is connected to the designated office Wi-Fi.
- **Requirement 4.1.3:** If not connected to the correct Wi-Fi, the user will see a message like, "Please connect to the office Wi-Fi to mark your attendance."
- **Requirement 4.1.4:** The attendance marking process should capture the employee's ID and the timestamp of the check-in.

4.2. Employee Features

- **Requirement 4.2.1: User Authentication:**
 - Employees will need a simple way to identify themselves, likely through an employee ID or a unique username.
 - For v1.0, a simple login with a pre-assigned Employee ID will be sufficient. No password is required initially to keep it simple, as the primary security is the Wi-Fi connection itself.
- **Requirement 4.2.2: Attendance Dashboard:**
 - Once logged in, the employee sees a clean dashboard.
 - It will display their name and employee ID.
 - It will clearly show the current date and time.
 - A prominent "Mark My Attendance" or "Check-in" button will be the main call to action.
 - The dashboard will show their attendance status for the current day (e.g., "Present since 09:15 AM" or "Not Marked Yet").
- **Requirement 4.2.3: Personal Attendance History:**
 - Employees can view a calendar or a list of their own attendance for the current month.

4.3. Administrator Features

- **Requirement 4.3.1: Admin Dashboard:**
 - Admins will have a separate, secure login.
 - The dashboard will show a real-time list of all employees who have checked in for

- the current day, along with their check-in times.
- It will also show a list of employees who have not yet checked in.
- **Requirement 4.3.2: Employee Management:**
 - Admins can add, edit, or remove employees from the system.
 - Each employee record will contain at least a Name and Employee ID.
- **Requirement 4.3.3: Attendance Reports:**
 - Admins can view attendance records for any given date range.
 - Admins can filter records by employee.
 - Admins can export the attendance data in a simple format like CSV.

5. Non-Functional Requirements

5.1. Performance

- The React application should be lightweight and performant, capable of running smoothly on a Raspberry Pi (e.g., Raspberry Pi 4 with 2GB/4GB RAM).
- The user interface should load quickly, especially the main check-in page for employees.

5.2. Security

- While the primary security is Wi-Fi based, the admin panel must be password-protected.
- The application should be accessible only within the local office network.

5.3. Usability

- The interface must be clean, minimal, and intuitive for non-technical users.
- The mobile experience should be excellent, as many employees might use their phones to mark attendance.

5.4. Technology Stack

- **Frontend:** React.js
- **Backend:** Node.js (Express.js) - running on the Raspberry Pi.
- **Database:** PostgreSQL - A powerful, open-source object-relational database system.
- **Host Environment:** Raspberry Pi OS with Node.js and PostgreSQL installed.

6. Future Considerations (Out of Scope for v1.0)

- **Check-out Feature:** Allowing employees to mark when they leave for the day.
- **Leave Management:** A module for employees to apply for leave and for admins to approve it.
- **Advanced Reporting:** More detailed graphical reports and analytics.
- **Profile Pictures:** Allowing employees to have profile pictures.
- **Integration:** Potential integration with other HR software via APIs.