Software Requirements Specification: Attendance Management System

REKELL MENDOZA CARAAN

A00133304

November 18, 2022

# Introduction

## Purpose

This software requirements specification describes the functional and non-functional requirements for software release 1.0 of the Attendance Management System (AMS). This document is intended to be used by the members of the project team who will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here are committed for release 1.0.

## Project Scope and Product Features

The AMS will allow two types of end users: the common end users and the privileged end users. Common end users will use AMS to capture or store their attendance to the current gathering or meeting that they have selected based on the available options. Most privileged end users will use AMS to view attendance reports and manage reminders or notifications intended to common end users belonging to their group or team. Few privileged end users will use AMS to setup and maintain the settings of a gathering or meeting.

# Overall Description

User Needs

Every congregation or association has their own regular member gatherings or meetings. Most of the time, monitoring and reporting of attendees and individual attendance per gathering/meeting is of most importance. Manually it can be done in paper or better by a spreadsheet. However, as technology emerges, platform agnostic application software is preferable and in demand. AMS addresses those needs in attendance management. End users can store and report their attendance on every gatherings/meetings with few clicks using their mobile devices. Other group of end users of AMS who have elevated permission can manage, view and report attendance of other end users under their group or team. Real-time reporting and accurate attendees’ reports are very attainable.

## Assumptions and Dependencies

* End users of AMS are all registered members of the congregation or association and must have email accounts (e.g. Gmail, etc.)
* AMS primarily depends on network connectivity. In the event of in-vicinity gathering (not remote), and there are no internet connections, AMS will just record the attendance of the end users in his/her local device. When back online, all records will be in-synced.

# System Features

Target Platforms and Back-end Server

The following are the target platforms or device for all end users:

* + - Android Devices (min Pixel 4 API 28)
    - IOS Devices (iPhone 14 IOS 16.1)
    - Windows Desktop
    - Mac

The following is the back-end server

* + - MS Azure Cloud
    - ASP.NET Core 6.0 Web App (Razor or Blazor)
    - MS Azure MSSQL (Database)

Process Flow

The following diagram below is the process flow of AMS.

Diagram

Description automatically generated

Server-Client Architecture

The following diagram below depicts the server and client architecture between the AMS client (e.g. Android, IOS, Windows, etc.) and AMS server residing in Microsoft Azure cloud. AMS client can only communicate to AMS Web Service if the end user of AMS client has a role of “Admin”. After authentication process, Admin end user can create, update, and delete a gathering/meeting from the AMS server. The AMS Web Service is a Razor/Blazor that listens to AMS client connection during gathering/meeting setup.

Diagram

Description automatically generated

The AMS Web Service will have the following main features:

|  |  |
| --- | --- |
| **FEATURES** | **DESCRIPTION** |
| Create a new gathering via Rest Api (“Post”) | This can be invoked either on the AMS Web App or via HttpClient invocation from the client device.  This http verb action will create a new gathering/meeting and be saved in the database. |
| Get the list of available gatherings via Rest Api (“GET”) | This can be invoked either on the AMS Web App or via HttpClient invocation from the client device.  This http verb action will get the list of available gatherings/meetings created and saved from the database. It will then be displayed on the client device calling this resource. |
| Update/Delete a gathering/meeting via Rest Api (“Delete”) | This can be invoked either on the AMS Web App or via HttpClient invocation from the client device.  This http verb action will delete/update a gathering/meeting from the list. |
| WebApp client | This is a web client user interface available only for Admin users of AMS. This enables the Admin user(s) to perform the CRUD operation of gatherings and attendees in/out of the database. |

The AMS Web Service is intended to be deployed and run in MS Azure cloud with a plan of using MSSQL or CosmosDB as a persistent storage or database.