Software Requirements Specification (SRS)

K. David Pearce

Project Step 9

MSSA San Antonio Cohort 5

Meets Standards Outlined in IEEE Std 830-1998:

IEEE Recommended Practice for Software Requirements Specification

Table of Contents

1. Introduction
   1. Purpose
   2. Scope
   3. Definitions, acronyms, and abbreviations
   4. References
   5. Overview
2. Overall Description
   1. Product perspective
   2. Product functions
   3. User characteristics
   4. Constraints
   5. Assumptions and dependencies
3. Specific Requirements

Introduction

1. Purpose
   a) The purpose of this SRS is to outline the functional requirements of the podcast website
   b) The intended audience of the SRS is the podcast producer and the web developer

2. Scope
   a) The software to be created is a web app for a podcast called Thank You For Your Service
   b) The product will have a user interface with podcast episodes, a link to sign up for emails, and a link to visit the online store. The product will also have a database for this data.

3. Definitions, Acronyms, Abbreviations – None

4. References
   a) This SRS was created following the guidelines in IEEE Std 830-1998

5. Overview
   a) The next sections will discuss the Overall Description and the Specific Requirements

Overall Description

1. Product Perspective
   a) There are no related products. The web app is self-contained.
   b) Users will need a browser and an internet connection to use the product.

2. Product Functions
   a) The primary function will be a internet address where users can stream podcast episodes.
   b) An alternate function will be an online store where users can purchase related items.

3. User Characteristics
   a) The intended users should know how to access the internet; no specific education level

4. Constraints
   a) Not every user will log in, so not every interaction will be traced

5. Assumptions and Dependencies
   a) The first version of this web app will not be mobile friendly

Specific Requirements

1. ASP.NET Core Web Application project
2. Add Controllers, Views, and Model
3. Link to a database
4. Setup interface to add content to database upon user input
5. Use Razor, HTML, and CSS to improve the user interface
6. Publish web app using Azure