Software Requirements Specification

for

Easy Issue Tracker

Version 1.0

Prepared by Filipe Lehmann Pereira

08/05/2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The software product specified in this document is the Easy Issue Tracker, made as a portfolio business project by Filipe Lehmann Pereira. The purpose and scope of it is to catalog and classify different issues created in the development of software projects by a single team. This document describes the system in its entirety.

1.2 Intended Audience and Reading Suggestions

This document is intended for developers, specially the project owner, and as a reference and content for the system itself. Each of the steps will be inserted and classified as the project progresses, after its main parts are functional. The rest of this SRS will describe the overall product functionalities and attributes as well as environment requirements and user interface.

1.3 Product Scope

The purpose of the software specified is to track, catalog, classify, assign and notify team members on the development of a software project in a way that facilitates the developer workflow.

2. Overall Description

2.1 Product Perspective

This product is its own self contained solution, based on other validated products and services already being executed. It is, although, a product that won't be marketed or sold and thus may offer the solution in a sub-optimal way.
It will be developed using Express.js as a back-end framework and React.js for the front-end.

2.2 Product Functions

The major functions of the project are:

1. Visualization of **Projects**, **Current User** and **Recently Viewed** issues in **Navbar**
    a. **Projects:** Selection of project and visualization of user's created issues
    b. **Current User**: User's created issues, name edition and logout
    c. **Recently Viewed:** Open in simple overlay if not accessed through **Current User**
2. Navigation in project through **Sidebar**, located on the left screen side
    a. Items: Home, Add Issue, Board and Project Settings
3. Configuration of issues in a Project Settings screen
    a. Settings of project name and members
        i. Assignment of registered users
        ii. Listing of members names, emails, roles, date of entry and remove option
        iii. (Consider option of filter by id, name or e-mail)

    b. Classification of issues by **type** in a custom ordered and colored manner.
    c. Classification of issues by **category** in a custom ordered manner.
4. Inclusion of issues in a specific screen and overlay modules
    a. Inclusion of issues in the **Add Issue** option in the **Sidebar**
    b. Category, Milestone and Version can be added via overlay modals
    c. Submission of Issue leads to message and link to created project edit page
5. Visualization of issues in a specific screen and overlay modules
    a. List display in bottom part with following columns
    b. Filters in top part
6. Visualization of board of issues
    a. Filters at top view section
    b. Board with 4 status swimlanes and its cards ordered by creation.

2.3 User Classes and Characteristics

The anticipated user classes are admin (or administrator) and user, with admin being the one with the ability to create projects and assign users to them. Both are able to use the main functions of the platform with the special functions described above available only to the administrator.

2.4 Operating Environment

The software will operate in a web environment, accessed through browsers like Google Chrome or Mozilla Firefox.

2.5 Assumptions and Dependencies

The possible IDEs used to develop the software are free or in the possession of the developer. The communication with a database provider will be free because of the non-commercial nature of the project. The technology used to develop the project is called MERN Stack, and will be explained in the system features section.

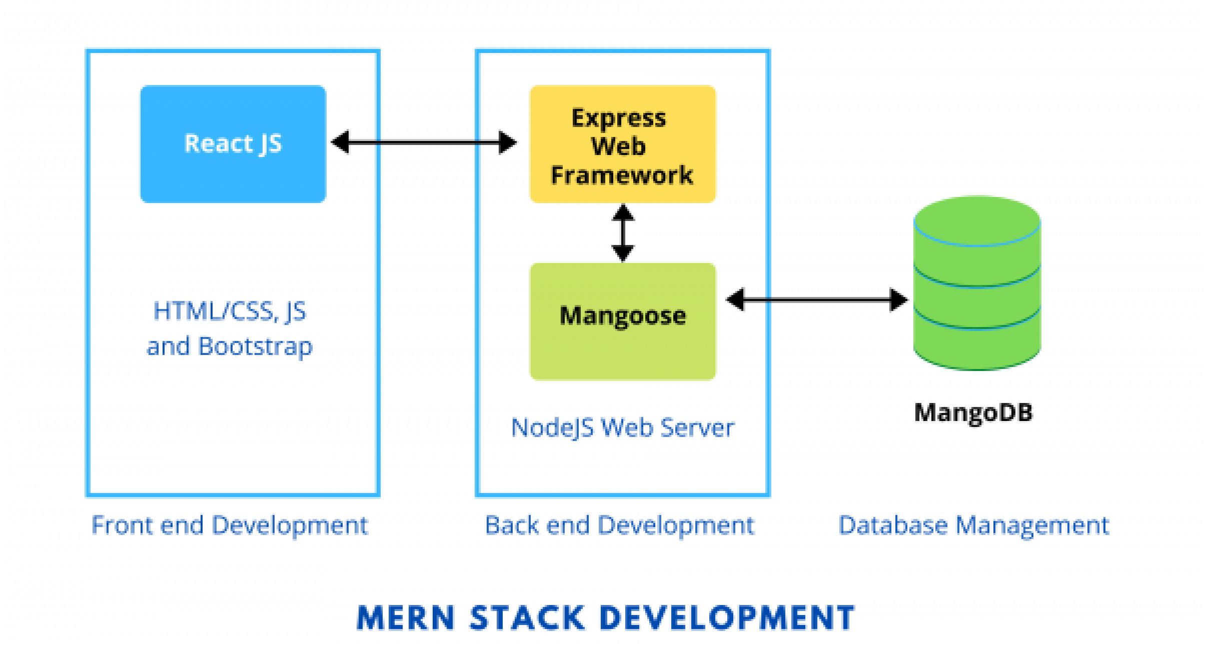
3. System Features

3.1 Description and priority

The Easy Issue Tracker will maintain data on projects and its cards, that represent issues or tasks to be done, as well as all the registered members, their rolls, their contributions and other useful and pertinent information.

3.2 MERN Stack

MERN stack is a collection of technologies to create scalable web applications. It’s an acronym for MongoDB, Express.js, React.js and Node.js. It can be divided into two blocks: a back-end and a front-end, with the backend communicating with the database which is managed separately, as can be seen in the following diagram.



4. External Interface Requirements

4.1 User Interfaces

The project will be developed with a mobile first approach, clean color design and React.js front end library in VS Code. The interface will be composed of 3 main sections and 4 subsections, listed below:

1. Login page (Authentication)
    a. Sign up page
2. Projects page / Dashboard
    a. Left, separate lists, loaded as needed
    b. Right, with statistics colored by status
3. Project Home
    a. Add Issues
    b. Issues
    c. Board
        i. Filters
        ii. Lanes
    d. Project Settings
        i. General
        ii. Issue Types
        iii. Categories

4.2 Hardware Interfaces

The final project needs a device with a browser with support to JavaScript and connection to the internet. The database of the application will be provided by MongoDB, a source-available cross-platform document-oriented database program, classified as a NoSQL database program.

4.3 Communication Interfaces

The communication in the project will occur between the database and a hosted website. It will happen using Express.js and Node.js which are the back-end technologies in the project.

5. Nonfunctional Requirements

5.1 Security Requirements

Services that handle personal or enterprise information and organization with database storage need security systems just like many other applications. In the Easy Issue Tracker case, all the requirements will be outsourced by MongoDB.

5.2 Software Quality Attributes

- Availability: Cards creation and update should be available at any time, so the workflow of the client is not disturbed.
- Correctness: The card description and attributes should be updated only by the members of the project and occupy its corresponding organizational positions in the system.
- Maintainability: The administrators and members in chargers should maintain members and team allocation in the projects.
- Usability: The boards should satisfy a maximum number of member needs.