Lab 2 – Traffic Tamer Product Specification

Antonio Qerozi, Anthony Akintunde, Joshua Pablo, Emily Reyna, Gavin Clendenin, Caitlyn Edwards, Aaron Edwards, Alexis Jones

CS411W

Professor Thomas J. Kennedy

November 25, 2024

Lab 2 Version 2

Table of Contents

1. Introduction

1.1 Purpose

This document outlines the requirements and design for **Traffic Tamer**, a web application created to help users understand and navigate traffic laws specific to different jurisdictions. The purpose of this document is to outline the requirements for **Traffic Tamer**, specifying its functional and non-functional requirements, and to provide a roadmap for development.

1.2 Scope

Traffic Tamer is designed to simplify traffic regulations for general drivers, individuals with traffic violations, driver improvement students, and law students, with the goal of providing clarified and easily accessible legal information to the user. Traffic Tamer helps users find and interpret traffic laws based on state, county, and specific violation codes. Utilizing machine learning, the application offers simplified legal explanations and personalized guidance. Its ultimate goal is to make traffic law education accessible and easy to understand.

1.3 Definitions, Acronyms, and Abbreviations

- **Apache**: Open-source web server software.
- **CSS**: Cascading Style Sheets, used for web page presentation.

- **Docker**: Platform for deploying applications in containers.
- **HTML**: Standard markup language for web pages.
- **JavaScript**: Programming language enabling interactive web pages.
- **SQL**: Open-source relational database.
- **Node.js**: runtime for server-side applications.
- **React**: JavaScript library for building user interfaces.
- **SQLite**: SQL database engine.
- **Traffic Law**: Regulations governing road users' behavior.
- **Express:** JavaScript Framework for simplified server-side communication.

1.4 References

Aghayari, H., Kalankesh, L. R., Sadeghi-Bazargani, H., & Feizi-Derakhshi, M.-R. (2021, August 2). Mobile applications for Road Traffic Health and safety in the mirror of the Haddon's

Matrix - BMC Medical Informatics and decision making. BioMed Central. Retrieved June 11, 2024, from https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-021-01578-8

Online legal services and legal advice. LegalShield. (2024, June 6). Retrieved June 11, 2024, from https://www.legalshield.com/

Traffic attorneys in North Carolina. iTicket.law - Powered by Hatley Law Office. (n.d.). Retrieved June 11, 2024, from https://www.iticket.law/

Traffic Caseload Highlights. (2024, April). National Center for State Courts. Retrieved June 11, 2024, from https://www.courtstatistics.org/__data/assets/pdf_file/0029/99920/Traffic-CLHL.pdf

Web Application Security: Risks, Technologies & Best Practices." CyCognito, Retrieved June 26, 2024, from www.cycognito.com/learn/application-security/web-applicationsecurity.php

Odnoletkov, Pavel. "Top 8 Web Application Security Threats and How to Mitigate Them." MBC Managed IT Services, 9 Aug. 2023, www.mbccs.com/web-application-security-threats/

"7 Common Web Security Threats for an Enterprise." Fortinet, Retrieved June 26, 2024, from www.fortinet.com/resources/cyberglossary/web-security-threats

Team Pearl. (2024, November 8). Lab 1 - Traffic Tamer Product Description.

Retrieved November 8, 2024 from https://cs-ereyn008.github.io/f24-Pearl/frontend/public/FAQ.html

1.5 Overview

Section 2 of this document offers a comprehensive overview of the Traffic Tamer system. It provides a detailed introduction to what the system is, its purpose, and its intended scope. It covers the primary features and functionalities that Traffic Tamer offers, gives readers an understanding of its capabilities, and describes the components that constitute the system to provide insight into its architecture. Section 2 also includes a discussion of the various user cases and scenarios in which Traffic Tamer is expected to be utilized, illustrating its practical applications and value to its users.

Section 3 delves into the specific functional and performance related requirements for the Traffic Tamer system. It outlines the exact operational capabilities that the system will provide to meet user needs and fulfill its objectives. Section 3 also specifies the performance standards that the system is expected to achieve in order to ensure its reliability and effectiveness during real-world use.

2. Overall Description

2.1 Product Perspective

Traffic Tamer is a web application designed to help users easily navigate and understand traffic law. It analyzes current traffic laws and checks regularly for updates to them. Through intelligent application of machine learning algorithms to this data, Traffic Tamer presents traffic laws based on the user's specified state, county, local police code, and violation code. This ensures users can easily and reliably find information relevant to them based on their specific circumstances.

The application’s ability to simplify complex laws into digestible terms will improve the user experience for those encountering difficulties understanding what they are reading. The application will offer automated law lookup based on violation code and a simplified explanation of said laws. Users seeking additional information can follow the application’s guided questions to further assess their situation, ensuring they understand the law and can make informed

decisions on their next steps. This tool aims to facilitate traffic law education for a broad user base, including drivers, traffic law students, and legal professionals.

2.2 Product Functions

Traffic Tamer allows users to search traffic laws and filter results by state and local codes. They receive straightforward explanations of complex regulations to allow them to navigate their legal situation more easily. This process is further enhanced through guided questions that provide more context to relevant laws, helping users stay informed on overall traffic legislation. Users are also able to create an account to bookmark frequently referenced laws.

2.3 User Classes and Characteristics

Traffic Tamer is designed to assist a wide range of users. **General drivers**, who operate vehicles on government roadways, benefit from the platform's ability to clarify complex traffic laws. **Drivers who have broken the law** and received violations also find value in Traffic Tamer, as it helps them understand the legal implications of their infractions and respond appropriately. For **driving students**, whether they are learning to drive for the first time or participating in driver improvement courses, the platform offers essential educational tools to enhance their understanding. Additionally, **law students** specializing in traffic law can rely on Traffic Tamer as a valuable resource for education and research, providing them with the insights they need for their studies and future practice.

2.4 Design and Implementation Constraints

N/A

2.5 Assumptions and Dependencies

- Users have basic internet access.
- Law data sources are reliable and regularly updated.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The application shall provide a web-based user interface accessible through standard web browsers on desktop and mobile devices.

Users will have access to:

- Account Creation and Login Page: To register and sign in, displaying fields for username, email, and password entry.
- Search Interface: Allowing users to select their state and input specific keywords to retrieve relevant traffic laws, with results showing summaries and links to the full text
- Simplification Feature: Available in the search results to provide users with layman's explanations of legal terms within the displayed traffic laws
- FAQ and Contact Pages: Accessible via the main menu to display frequently asked questions and provide contact

3.1.2 Hardware Interfaces

The application does not require specialized hardware interfaces; it is accessible on any device with internet connectivity and a compatible web browser.

The system will operate on servers configured to support web hosting and data storage (e.g., cloud-based infrastructure).

3.1.3 Software Interfaces

The frontend will utilize React or Angular frameworks to enable a responsive user experience across desktop and mobile devices.

The backend will employ Node.js for server-side processing, SQLite for database management, and Docker for deployment across various environments.

The application will integrate with external resources for traffic law data, using APIs to retrieve the latest traffic laws and updates.

3.1.4 Communications Interfaces

The application shall communicate with external APIs to retrieve and update traffic law data from reliable sources, ensuring users have access to the most current information.

HTTPS protocol will be used to secure data transmission between users and the server, ensuring the protection of user login information and search queries.

User interactions, including search and bookmark functionalities, will be logged to enhance performance monitoring and security tracking.

3.2 System Features

3.2.1 Account Management

3.2.1.1 Introduction/Purpose of Feature

This feature allows the capability to create accounts on the Traffic Tamer system granting the users access to the features of the application and allows the associated actions that can be done with the account such as account recovery.

3.2.1.2 Stimulus/Response

Stimulus: A request is done to create an account by providing their name, email address, location, and a password

Response: The provided information is processed through the system and stores the details of the account within the database.

3.2.1.3 Associated Functional Requirements

3.2.1.3.1 User Registration Page (O: Pablo, M1: Reyna)

The system shall present a page for the users to input a name, email address, location, and password for their account creation.

3.2.1.3.2 Login Page (O: Pablo, M1; Reyna)

The system shall possess a page that allows users to login with their email address/username and password.

3.2.1.3.3 Account Recovery (O: A. Edwards, M1: C. Edwards, M2: Pablo)

The system shall allow users to recover lost usernames through a presented page that prompts an input for an email address.

The system shall allow users to recover lost passwords through a presented page that prompts an input for a username or email address.

3.2.2 Application Information and Assistance

3.2.2.1 Introduction/Purpose of Feature

The application information and assistance feature are to provide users with the ability to search for traffic laws and information on how the platform works. It also gives users resources to utilize if they have questions or issues with the application.

3.2.2.2 Stimulus/Response

Stimulus: The user enters the respective information for a traffic ticket they have received on the ticket search page.

Response: The system outputs the violation the traffic ticket is associated with and provides the legal text, a simplified explanation, recommendations on avoiding repeat violations, and questions on how the user understands the traffic law.

3.2.2.3 Associated Functional Requirements

3.2.2.3.1 Ticket Search (O: Pablo, M1: Reyna)

The system shall have a page that possesses a form the user will input with the appropriate information of their traffic ticket.

3.2.2.3.2 Search Functionality (O: Jones, M1: Jones)

The system shall allow users to search for traffic laws by selecting a specific U.S. state from a predefined list or drop-down menu.

3.2.2.3.3 Search Results (O: Pablo, M1: Reyna)

Upon a chosen traffic law or validated traffic ticket, the system shall display the violation or traffic law with the appropriate legal text, simplified explanation, and recommendations on avoiding the violation.

3.2.2.3.4 Frequently Asked Questions Page (O: A. Edwards, M1: Reyna)

The system shall possess a FAQ information page for user awareness and clarify some details about the platform.

3.2.2.3.5 Contact Page (O: A. Edwards, M1: Reyna)

The system shall be able to display a list of contact information.

3.2.3 User Feedback Integration

3.2.3.1 Introduction/Purpose of Feature

This feature is to allow users to provide back critical feedback on the system to provide insight as to how the platform could be improved upon or what features are done well.

3.2.3.2 Stimulus/Response

Stimulus: A user reports a law to be either inaccurate or provides a misleading simplified explanation.

Response: The system records the user's report and is stored in the database for later viewing by the development team.

3.2.3.3 Associated Functional Requirements

3.2.3.3.1 Feedback Section (O: Reyna, M1: Reyna)

The system shall allow users to provide feedback on the accuracy and clarity of traffic law information presented in search results.

3.2.3.3.2 Compilation of Feedback (O: Reyna, M1: Reyna)

The system shall compile user feedback to continuously improve the simplification algorithm and content accuracy.

3.2.3.3.3 Administrator Review (O: Reyna, M1: Reyna)

The system shall have a section for administrators will them to have access to feedback analytics to monitor areas needing clarification or improvement.

3.2.4 Backend Information Processing

3.2.4.1 Introduction/Purpose of Feature

The purpose of backend information processing is to manage the traffic law content, user accounts and login authentication.

3.2.4.2 Stimulus/Response

Stimulus: A user interacts with the website

Response: The backend system communicates with the database to retrieve appropriate information to serve to the front-end.

3.2.4.3 Associated Functional Requirements

3.2.4.3.1 User Authentication (O: A. Jones, M1: Akintunde)

The system shall authenticate users via email and password, securely hash passwords with password-specific hashing algorithm, and save user metadata in a database.

3.2.4.3.2 Data Retrieval (O: A. Jones)

The system shall retrieve user data from the database based on user email with a response time of less than 2 seconds.

3.2.4.3.3 Form Submission Handling (O: A, Jones, M1: Akintunde)

The system shall validate and sanitize all ticket information submissions, ensure protection against SQL injection attacks, store data securely in the database, and provide appropriate success or error messages to the user.

3.2.4.3.4 API Endpoint Creation (O: A. Jones, M1: Akintunde)

The system shall provide RESTful API endpoints supporting the full CRUD operations for user data ensuring each endpoint is accessible, functional and secure, endpoints shall include:

- GET /users - Retrieve a list of users.

- POST /users - Create a new user.
- GET /users/{id} - Fetch specific user data using user id.
- DELETE /users/{id} - Remove a user via id.
- PUT /users/{id} - Update an existing user.

3.2.4.3.5 Error Logging (O: A. Jones, M1: Akintunde)

The system shall log errors with a timestamp, severity level, and an error description to a dedicated log file.

3.2.4.3.6 Data Validation (O: A. Jones, M1: Akintunde)

The system's backend shall validate and sanitize data for each input field according to predefined rules, ensuring that required fields are completed, formatted correctly, and free from malicious input.

3.2.4.3.7 Notification System (O: A. Jones, M1: Akintunde M2: Clendenin)

The system shall send email notifications to users for important events, such as account creation, password changes, and traffic law updates on bookmarked laws.

3.2.4.3.8 Data Encryption (O: A. Jones M1: Akintunde M2: Clendenin)

All passwords, personal identification information, and other potentially sensitive or important information in the system shall be encrypted to protect user information.

3.2.4.3.9 Search Functionality (O: A. Jones, M1: Akintunde)

The system's backend shall support searching user data by criteria such as email or registration date and return results within 2 seconds.

3.2.4.3.10 Data Backup (O: A. Jones)

The system shall perform automated backups of the user database every 24 hours.

3.2.4.3.11 Session Management (O: A. Jones)

The system shall manage user sessions with a timeout feature, automatically logging out users after 15 minutes of inactivity.

3.2.4.3.12 Performance Monitoring (O: A. Jones)

The system's backend shall track API response times, logging any request that takes longer than 3 seconds for performance analysis.

3.2.4.3.13 Role-Based Access Control (O: A. Jones M1: Clendenin)

The system's backend shall implement role-based access, granting specific permissions based on user roles (user, guest, admin).

3.2.4.3.14 Audit Trail (O: A. Jones M1: Clendenin)

The system shall maintain a trail of user actions (logins or modifications) for accountability and compliance purposes.

3.2.4.3.15 Machine Learning Processing (O: Akintunde M1: Clendenin)

The system's backend shall use a machine learning algorithm to simplify provided traffic law data into clear, easy-to-understand language for users.

3.2.5 Database System

3.2.5.1 Introduction/Purpose of Feature

The database system features will be used to keep track of user credentials and searches, giving users the ability to look back upon their previous searches/explanations and store information shared with the application.

3.2.5.2 Stimulus/Response

Stimulus: The user enters their information within the application, including the violation and circumstances of the violation.

Response: The system will store the information for a set time, up to a year, giving users the ability to backtrack and look at their previous searches/situations.

3.2.5.3 Associated Functional Requirements

3.2.5.3.1 System Search Storing (O: A. Qerozi M1: Akintunde)

The system shall store and display the user's previous searches for up to a year before deleting.

3.2.5.3.2 User Credential Storing (O: A. Qerozi, M1: Akintunde)

The system shall store user login credentials, such as email and encrypted password, for an indefinite period, allowing users to have access to their accounts forever once created.

3.2.6 Backend Bookmark Storage

3.2.6.1 Introduction/Purpose of Feature

The backend bookmark storage feature's purpose is to allow users to easily save and manage links to frequently referenced pages and resources on the website.

3.2.6.2 Stimulus/Response

Stimulus: A user is visiting the traffic tamer website and pulls up a law. Then they press the bookmark button to come back to it later.

Response: The system makes note of the bookmark and the user who book marked it. The bookmark is saved to that user's profile.

3.2.6.3 Associated Functional Requirements

3.2.6.3.1 Bookmark Information Storing (O: A. Jones, M1: Jones)

The system shall store user bookmark information.

3.2.7 Backend Logging

3.2.7.1 Introduction/Purpose of Feature

The backend logging feature helps to maintain and improve performance, security, and user experience of the Traffic Tamer website. It involves recording events and interactions within the backend structure to provide valuable insights into how the application operates. It ensures the application remains reliable, secure, and adaptable to user needs over time.

3.2.7.2 Stimulus/Response

Stimulus: A user tries to access information on specific traffic law, but the page takes too long to load and eventually displays an error message.

Response: The backend logging system records the error, noting the slow response time and capturing the details of the query being run, the server response code, and the time of the request.

3.2.7.3 Associated Functional Requirements

3.2.7.3.1 Action Logging (O: A. Jones, M1: Jones)

The system shall log important actions (user logins, account creations, etc.) to track the system's performance and security.

3.3 Performance Requirements

3.3.1 Processing Speeds and Loading

3.3.1.1 Ticket and Traffic Law Search Time (O: Pablo)

The average time for a traffic ticket or traffic law search up time shall not exceed 10 seconds.

3.3.1.2 Account Management Time (O: Pablo)

The average time for registration of an account and account recovery upon a user's submission of information shall be within 20 seconds.

3.4 Design Constraints

3.4.1 Compatibility

3.4.1.1 Web Browser Compatibility (O: Pablo)

The Traffic Tamer Platform must be compatible with the most common web browsers used.

3.4.1.2 Device Compatibility (O: Pablo)

The application must be compatible with various devices and operating systems such as Windows and mobile devices.

3.5 Software System Attributes

3.5.1 Reliability

The system must have an accuracy of at least 95% in its traffic law accuracy and simplification.

3.5.2 Availability

The system shall be constantly available and only be made unavailable when the system has to undergo maintenance or recovery.

3.5.3 Security

User data and information must be encrypted and secured with an encryption algorithm such as AES in order to maintain user integrity and privacy.

3.5.4 Maintainability

The system shall follow standard programming principles, be well-documented, and be analyzed monthly for consistent maintainability.