UNIVERSIDAD DE GUADALAJARA

Centro Universitario de los Valles

Maestría en Ingeniería de Software

Software Configuration Management Project

Virtual Assistant for Recognizing Dangerous Roads.

Professor:
Dr. Omar Ali Zataraín Durán
Student:
Rodrigo Reséndiz Ávila 222977423

Master in software engineering
August 18, 2023
Ameca, Jalisco

Contenido

1. Introduction

Effective software management is paramount in today's environment, where technology plays a crucial role in the development of innovative solutions. Proper project management ensures not only the timely delivery of quality products but also resource optimization and customer satisfaction. In this context, we will step through the development of a project worked on in class, highlighting key aspects of software management and showcasing practical applications in real-world scenarios. This project stands out for its focus on developing advanced solutions, including key change requests that address shifting client demands and government regulations. Through a robust software management strategy, we aim not only to meet technical objectives but also to maximize delivered value, ensuring the security, efficiency, and comprehensive functionality of the system. In this exploration, we will delve into the critical importance of software management and its practical application, using our classroom project as a guiding example.

2. Baseline

Before continuing with Software Configuration Management (SCM) a baseline must be designed with which the development of software administration can begin.

A Baseline in software development is a baseline that establishes the fundamental elements of the project at a given time. It contains crucial information that serves as a reference for future comparisons and evaluations. Baselines can be updated to reflect changes and evolutions in the development process.

The baseline may contain the following points:

- **Project description:**
  - General objective and specific objectives.
  - Project scope, including key modules or features.
- **User Requirements:**
  - Detailed list of functional and non-functional requirements.
  - Prioritization of requirements according to importance and criticality.
- **System architecture:**
  - Description of the general architecture of the system.
  - Specification of technologies and tools to be used.
- **User Interface Design:**
  - Mockups or prototypes of user interfaces.
  - Design specifications, including usability principles.
- **Work Breakdown and Schedule:**
  - Time estimation and resource allocation
  - List of tasks and activities

- **Risk management:**
  - Identification of potential risks.
  - Mitigation strategies and contingency plans.
- **Criteria of acceptance:**
  - Conditions that must be met for the project to be considered successful.
  - Quality criteria and tests.
- **Human Resources:**
  - List of roles and responsibilities.
  - Assignment of personnel to each role.
- **Budget:**
  - Detail of estimated costs.
  - Budget distribution by category (development, testing, resources, etc.).
- **Communication and Collaboration:**
  - Tools and communication channels.
  - Frequency and formats of reports and updates.

For the project, 8 modules were adjusted, each described with its respective functional and non-functional requirements.

Main features:

Module 1: Data Acquisition

- Functional Requirements:

1.1. The system must be capable of collecting data from multiple sources, such as traffic cameras, vehicle sensors, and real-time weather data.
1.2. The system must efficiently store and manage large volumes of data.
1.3. It should be able to sync and merge data from various sources to gain a comprehensive view of road conditions.

- Non-Functional Requirements:

1.4. The system must ensure the security and privacy of collected data.
1.5. System performance must be scalable to handle increases in data volume.
1.6. Data processing latency must be low to enable real-time responses.

Module 2: Data Processing

- Functional Requirements:

2.1. The system must apply computer vision and machine learning algorithms to identify dangerous roads.
2.2. It should analyze traffic speed, weather conditions, and other relevant factors.
2.3. It should generate alerts when hazardous conditions are detected.

- Non-Functional Requirements:

2.4. The system must process and analyze data in real-time.
2.5. Detection accuracy of dangerous roads must be high.
2.6. The system should be adaptable to various weather and lighting conditions.

Module 3: Alert Generation

- Functional Requirements:

3.1. The system must send alerts to drivers through a user-friendly interface.
3.2. It should provide specific recommendations for safe driving.
3.3. It should be able to communicate with navigation systems and voice assistants.

- Non-Functional Requirements:

3.4. Alerts must be clear and understandable.
3.5. The system should have redundancy to ensure critical alerts are delivered.
3.6. The speed of alert delivery must be minimal.

Module 4: Monitoring and Maintenance

- Functional Requirements:

4.1. The system must monitor the operational status of key components like cameras and sensors.
4.2. It should generate performance and preventive maintenance reports.

- Non-Functional Requirements:

4.3. The system must be capable of performing remote software updates.
4.4. It should have high availability to prevent service disruptions.

Module 5: Administration Interface

- Functional Requirements:

5.1. It should provide an administration interface for configuring and customizing the system.
5.2. It should allow access to historical data and trend analysis.
5.3. It must be secure and require authentication for access.

- Non-Functional Requirements:

5.4. The administration interface should be intuitive and user-friendly.
5.5. It should be compatible with multiple browsers and devices.

Module 6: Vehicle Integration

- Functional Requirements:

6.1. The system must integrate with vehicle information and entertainment systems.
6.2. It should provide real-time feedback to drivers through the vehicle's interface.
6.3. It must allow bidirectional communication to receive vehicle telemetry data.

- Non-Functional Requirements:

6.4. Vehicle integration should be compatible with a variety of makes and models.
6.5. Latency in communication with the vehicle should be minimal for a real-time driving experience.

Module 7: Notifications to Authorities and Emergency Services

- Functional Requirements:

7.1. The system must be capable of sending automatic notifications to traffic authorities and emergency services when extremely hazardous conditions are detected.
7.2. It should include detailed information about the location and nature of the danger.

- Non-Functional Requirements:

7.3. Notifications to authorities must be reliable and delivered in real-time.
7.4. The system must comply with regulations and standards for communication with authorities and emergency services.

Module 8: Historical Data Analysis and Report Generation

- Functional Requirements:

8.1. The system must be capable of storing and analyzing historical data regarding hazardous road conditions.
8.2. It should generate periodic reports and trend analysis to enhance long-term road safety.
8.3. It should provide data visualization tools for authorized users.

- Non-Functional Requirements:

8.4. The system must be efficient in processing historical data.
8.5. Report generation should be configurable and customizable to meet user needs.
8.6. It must ensure the security and privacy of historical data.

This is a first delivery of the baseline. it is not perfect but with what will be seen soon improvements can be made in each version.

3. Configuration Management

Configuration management is a discipline applying technical and administrative direction and surveillance over the life cycle of item to: [1]

- Identify and document the functional and physical characteristics of configuration items (Cis).
- Control changes to these characteristics.
- Verify conformance with specified requirements.
- Record and report change processing and implementation status.

3.1 Configuration identification

The configuration identification is the process of defining and identifying configuration elements in a system or software project. These configuration items can include hardware, software, documents, data, and any other components relevant to the system. Configuration identification is a crucial step in configuration management, as it establishes the basis for controlling and tracking elements throughout the software development lifecycle.[2]
The main objectives of configuration identification are:

- Definition of Configuration Elements (CI): Identify and clearly define each element that contributes to the system. This may include source code, design documents, databases, hardware configurations, among others.

- Establishing Unique Identifiers: Assign unique identifiers to each configuration item to facilitate traceability and tracking. These unique identifiers are often managed by version control systems.

- Creating Baseline Configurations: Determine and establish specific system configurations at key moments in development, known as "baselines."

- Configuration Documentation: Create detailed documentation that describes each configuration item, including its purpose, function, associated versions, and any dependencies on other items.

- Change Tracking: Facilitate change management by providing a framework to record and track changes to configuration items over time.

3.2 Configuration change control

Configuration Change Control is a process within configuration management that evaluates, approves, or rejects proposed changes to the configuration items in a system or software project. This process is implemented to ensure that any modifications to the system are managed in a controlled manner, minimizing risks, and maintaining the integrity and stability of the software.

Key activities associated with Configuration Change Control include:

- **Change Request**: A change may be proposed for various reasons, such as bug fixes, feature enhancements, or adaptation to new requirements. Anyone on the development team or even end users can propose a change.

- **Change Evaluation**: The change request is evaluated by a change control board or designated individuals. Aspects such as technical feasibility, impact on scope, timelines, and budget, and any other relevant factors are analyzed.

- **Decision Making**: Based on the evaluation, a decision is made to approve, reject, or defer the proposed change. The decision may depend on the priority of the change, its impact on the system, and the availability of resources.

- **Change Implementation:** If the change is approved, its implementation is carried out. This may include updating source code, documents, configurations, and other affected elements.

- **Testing and Validation**: After implementing the change, testing and validation are performed to ensure there are no unintended side effects and that the system continues to meet requirements.

- **Documentation Update**: Documentation, including configuration documentation, is updated to reflect the implemented change.

Configuration Change Control is essential for maintaining the stability and consistency of the system over time. It provides a framework for effectively managing changes, reducing risks associated with unplanned or poorly managed modifications. Proper traceability and documentation are critical components of this process to ensure the integrity of the system and facilitate audits and reviews.

3.3 Configuration Audits

A Configuration Audit is a systematic examination of the configuration items and associated documentation to ensure that they conform to their specified requirements. This process is an essential component of configuration management and is conducted to verify the completeness, consistency, and correctness of the configuration items within a system or project.

Key objectives of configuration audits include:

- Verification of Compliance: Ensuring that the configuration items adhere to the specified requirements, including functional, performance, and design specifications.

- Identification of Discrepancies: Detecting and resolving any inconsistencies, discrepancies, or deviations from the established baseline or standards.

- Documentation Accuracy: Verifying the accuracy and completeness of configuration documentation, including version control, change history, and relationships between different items.

- Change Control Assessment: Assessing the effectiveness of the change control process to determine whether changes have been properly reviewed, approved, and implemented.

There are different types of configuration audits, including:

- Functional Configuration Audit (FCA): Focuses on verifying that the system's functional and performance characteristics meet the specified requirements.

- Physical Configuration Audit (PCA): Concentrates on the examination of the physical configuration, including hardware and software components, to ensure they match the design documentation.

- System Configuration Audit: Encompasses a comprehensive review of the entire system configuration, combining elements of both FCA and PCA.

The configuration audit process is typically carried out by an independent group, often referred to as a Configuration Audit Board or Configuration Control Board, to maintain objectivity and impartiality. Audits may be conducted at various stages of the project lifecycle, including during development, testing, and post-implementation phases. The results of configuration audits contribute to ensuring the quality, integrity, and reliability of the configured items in a system or project.

3.4 Configuration status accounting

Configuration Status Accounting (CSA) is a key component of Configuration Management that involves the systematic recording and reporting of the state and history of the configuration items (CIs) throughout their lifecycle. The primary goal of Configuration Status Accounting is to provide accurate and up-to-date information about the configuration items, their versions, and their relationships.

Key aspects of Configuration Status Accounting include:

- Recording Changes: Documenting any changes made to the configuration items, including updates, modifications, and new additions. This includes changes to both hardware and software components.

- Version Control: Maintaining a version history for each configuration item, allowing for traceability and understanding of how the item has evolved over time. This includes recording the creation, modification, and deletion of items.

- Baseline Management: Documenting the establishment of baselines, which represent specific, approved configurations at particular points in time. Baselines serve as reference points for configuration comparisons and change control.

- Traceability: Establishing and maintaining traceability links between different configuration items, ensuring that relationships between various elements are well-documented.

- Reporting: Generating reports that provide insights into the current state and history of configuration items. These reports may include configuration summaries, change histories, and status reports.

- Audit Trail: Creating an audit trail that allows for the tracking of who made changes, when changes were made, and the reasons behind those changes. This is crucial for accountability and compliance.

Configuration Status Accounting is often facilitated by configuration management tools that automate the recording and tracking of configuration information. This information is valuable for project managers, developers, and other stakeholders to make informed decisions about the status, changes, and overall health of the configuration items within a system or project.

Centro Universitario de los Valles

Maestría en Ingeniería de Software

Software Configuration Management Project

BaseLine Virtual Assistant for Recognizing Dangerous Roads.

Professor:
Dr. Omar Ali Zataraín Durán
Student:
Rodrigo Reséndiz Ávila 222977423

Master in software engineering
Sept 22, 2023
Ameca, Jalisco

1. Control Version

Table 1.1 Presentation of the first project baseline.

Date	06/09/2023	Document	BaseLine_RodrigoResendiz
Configuration Item Description	Introduction	Initial release	
	General Description	Initial release	
	Requirements	Initial release	
Documented on the repository of the project			Yes
Previous document version		None	
Comments			
No Comments			

Table1.2 Update baseLine

Date	28/11/2023	Document	BaseLine_V1.1
Configuration Item Description	The baseline is described according to the points mentioned in this work.		
Documented on the repository of the project			Yes
Previous document version		BaseLine_RodrigoResendiz	
Comments			
All missing points are added and the project is described in more detail.			

2. Introduction

In the field of technological innovation, the "Virtual Assistant for the Recognition of Dangerous Roads" project emerges as a visionary company with significant implications in improving road safety. This project aims to develop an advanced virtual assistant, capable of using pattern recognition and deep learning algorithms to detect and alert about dangerous conditions in real time on our roads.

Prior to the deployment of this initiative, it is imperative to establish a solid Baseline. This reference point will not only define the fundamental elements of the project, but will also act as the cornerstone for the management, evaluation and continued success of the development. From the technical specification of the modules to the rigorous definition of functional and non-functional requirements, this Baseline will be the compass that will guide each iteration of the project towards its completion.

.

3 Abstract

Any car owner will be able to access the assistant; all they need to do is activate it to begin its operation. This means that the assistant will be able to visualize the

user's route in real-time, check for optimal speed, and monitor the time it takes for the journey. It will utilize artificial intelligence to analyze the type of route taken, road conditions, and incline. Additionally, it will have the capability to send alerts to other users as well as emergency services in the event of an accident, utilizing geolocation within at least a 10km radius.

To implement this, machine learning techniques will be employed along with speed sensors, gyroscopes, accelerometers, and cameras. These will be used to gather information, which will then be stored in a database for processing. This will enable the assistant to generate visual or auditory alerts, as well as provide alternate route recommendations.

4 Project baseline description

4.1 **General Objective**:

The overarching aim of this innovative project is to enhance road safety through the development and implementation of an advanced Virtual Assistant tailored for the recognition of hazardous conditions on roads. By leveraging cutting-edge technologies, including pattern recognition and artificial intelligence, the project seeks to create a robust system capable of identifying and alerting users to potential dangers in real-time.

4.2 **Objective**

- Develop a sophisticated algorithm for the real-time recognition of hazardous road conditions.
- Implement a user-friendly interface for seamless interaction with the Virtual Assistant.
- Integrate machine learning capabilities to enhance the system's ability to adapt and improve hazard recognition over time.
- Establish a scalable and modular architecture to accommodate future updates and expansions.

4.3 Project Scope:

The project's scope encompasses the following key modules and features:

- Data Acquisition
- Data Processing
- Alert generation
- Monitoring and maintenance
- Administration Interface
- Vehicle integration
- Notifications to authorities and emergency services
- Historical Data Analysis and Report Generation

The features are described in the following point.

4.4 Requirements

Functional and Non-Functional Requirements play an essential role in charting the path towards creating effective technological solutions. Functional Requirements outline the specific capabilities and functions that the software must meet, while Non-Functional Requirements define the characteristics that affect its performance, security, and user experience.

Having these requirements clearly defined from the beginning is like mapping out a journey before embarking on it. Functional Requirements act as specific targets to be achieved, setting expectations for what the software must achieve to satisfy the user's needs. On the other hand, Non-Functional Requirements are the conditions that guarantee that this trip is smooth, efficient and safe.

4.4.1 Functional Requirements

Module 1	Data Acquisition	RF-01	The system must be capable of collecting data from multiple sources, such as traffic cameras, vehicle sensors, and real-time weather data.
		RF-02	The system must efficiently store and manage large volumes of data.
		RF-03	It should be able to sync and merge data from various sources to gain a comprehensive view of road conditions.
Module 2	Data Processing	RF-04	The system must apply computer vision and machine learning algorithms to identify dangerous roads.
		RF-05	It should analyze traffic speed, weather conditions, and other relevant factors.
		RF-06	It should generate alerts when hazardous conditions are detected.
Module 3	Alert Generation	RF-07	The system must send alerts to drivers through a user-friendly interface.
		RF-08	It should provide specific recommendations for safe driving.
		RF-09	It should be able to communicate with navigation systems and voice assistants.
Module 4	Monitoring and Maintenance	RF-10	The system must monitor the operational status of key components like cameras and sensors.
		RF-11	It should generate performance and preventive maintenance reports.

Module 5	Administration Interface	RF-12	It should provide an administration interface for configuring and customizing the system
		RF-13	It should allow access to historical data and trend analysis.
		RF-14	It must be secure and require authentication for access.
Module 6	Vehicle Integration	RF-15	The system must integrate with vehicle information and entertainment systems.
		RF-16	It should provide real-time feedback to drivers through the vehicle's interface
		RF-17	It must allow bidirectional communication to receive vehicle telemetry data.
Module 7	Notifications to Authorities and Emergency Services	RF-18	The system must be capable of sending automatic notifications to traffic authorities and emergency services when extremely hazardous conditions are detected.
		RF-19	It should include detailed information about the location and nature of the danger
Module 8	Historical Data Analysis and Report Generation	RF-20	The system must be capable of storing and analyzing historical data regarding hazardous road conditions.
		RF-21	. It should generate periodic reports and trend analysis to enhance long-term road safety.

4.4.2 Non-Functional Requirements

RNF-01	The system must ensure the security and privacy of collected data.
RNF-02	System performance must be scalable to handle increases in data volume.
RNF-03	Data processing latency must be low to enable real-time responses
RNF-04	Detection accuracy of dangerous roads must be high.
RNF-05	The system should be adaptable to various weather and lighting conditions.
RNF-06	Alerts must be clear and understandable.
RNF-07	The system should have redundancy to ensure critical alerts are delivered.
RNF-08	The speed of alert delivery must be minimal.

RNF-09	The system must be capable of performing remote software updates.
RNF-10	The administration interface should be intuitive and user-friendly.
RNF-11	It should be compatible with multiple browsers and devices.
RNF-12	Vehicle integration should be compatible with a variety of makes and models.
RNF-13	Latency in communication with the vehicle should be minimal for a real-time driving experience
RNF-14	Notifications to authorities must be reliable and delivered in real-time
RNF-15	The system must comply with regulations and standards for communication with authorities and emergency services.
RNF-16	The system must be efficient in processing historical data
RNF-17	Report generation should be configurable and customizable to meet user needs.
RNF-18	It must ensure the security and privacy of historical data

4.5 System Architecture:

The architecture of the Virtual Assistant for Hazardous Road Recognition is designed to be robust, scalable, and adaptable to various environments. The system is divided into key components that seamlessly work together to ensure efficient hazard detection and user interaction.

Image Recognition Module	Utilizes advanced image processing algorithms to analyze real-time data from road-facing cameras.
	Employs computer vision techniques to identify and categorize potential hazards, such as potholes, construction zones, and adverse weather conditions.
Sensor Integration:	Integrates data from a variety of sensors, including GPS for location tracking and weather sensors for real-time weather conditions.
	Enables the system to correlate sensor data, providing a

	comprehensive view of the road environment.
Machine Learning Engine:	Implements machine learning models to continuously improve hazard recognition accuracy.
	Adapts and refines hazard detection based on historical data, user feedback, and evolving road conditions.
User Interface (UI):	Offers an intuitive and responsive UI accessible via both mobile applications and in-vehicle displays.
	Provides real-time hazard alerts with visual indicators and audible warnings.
	Includes features for user feedback on hazard accuracy and system performance.
Backend Infrastructure:	Establishes a robust backend infrastructure to handle data processing, storage, and retrieval.
	Implements cloud-based solutions for scalability, ensuring optimal performance during peak usage.

4.5.1 Specification of Technologies and Tools:

Programming Languages:	Python for backend development and algorithm implementation.
	JavaScript and HTML/CSS for front-end development.
Frameworks and Libraries	OpenCV for computer vision and image processing.
	TensorFlow or PyTorch for machine learning model development.
	Flask or Django for backend web framework.
Database	MongoDB or PostgreSQL for efficient data storage and retrieval.
Cloud Services	Integration with cloud services such as AWS or Azure for scalable infrastructure.

User Interface Development	ReactJS or Angular for building responsive and interactive user interfaces.
Communication Protocols	RESTful APIs for seamless communication between frontend and backend components.
Version Control	Git for version control, ensuring collaborative and organized development.

4.6 Work Breakdown and Schedule

Name	Time	Start Time	Finish Time
Collect process information.	30 days	18/08/2023	19/09/2023
Specification of requirements.	60 days	18/08/2023	19/10/2023
Technology selection	30 days	18/08/2023	19/09/2023
Development of the workspace	60 days	18/08/2023	19/10/2023
Vehicle Integration	90 days	08/10/2023	08/01/2024
Create interface	90 days	09/09/2023	09/12/2023
Create alert	60 days	09/09/2023	09/11/2023
Create database	90 days	09/10/2023	09/01/2024
Create intercommunication	90 days	09/09/2023	09/12/2023
Development notification and alert	90 days	09/09/2023	09/12/2023
Development processing data	90 days	01/01/2024	01/04/2024
Generate Manuals and maintenance	60 days	01/04/2024	01/06/2024
Report result	30 days	01/06/2024	01/07/2024
Error corrections	60 days	01/07/2024	01/09/2024

4.7 Risk management:

- Identification of Potential Risks:

Name	Risk	Mitigation Strategy	Contingency plan
Algorithmic Accuracy	Inaccuracies in hazard detection algorithms may lead to false positives or negatives.	Conduct rigorous testing with diverse datasets and road conditions to enhance algorithm accuracy.	Implement a user feedback mechanism to quickly identify and correct algorithmic errors.
Data Security and Privacy	Potential breaches of data security and privacy concerns related to user information.	Implement robust encryption protocols and comply with data protection regulations.	Establish a response plan for immediate action in case of a security incident, including user notification and system updates.
Integration Challenges	Difficulties in integrating the Virtual Assistant with various vehicle systems and platforms.	Collaborate closely with automotive manufacturers and ensure compatibility during development.	Develop adaptors or APIs for seamless integration with different vehicle models, minimizing potential disruptions.
User Adoption	Users may be resistant to adopting the Virtual Assistant, impacting its effectiveness.	Implement an extensive user education and onboarding process, highlighting the benefits and ease of use.	Gather user feedback continuously and adjust the user interface or educational materials based on user insights.
Regulatory Compliance	Changes in road safety regulations may impact the legality of the Virtual Assistant's operation.	Stay informed about and proactively adhere to regulatory requirements throughout development.	Establish a legal team to monitor and address changes in regulations, ensuring ongoing compliance

4.7.1 Mitigation Strategies and Contingency Plans:

Name	Mitigation strategy	Contingency plan
Continuous Testing and Validation	Implement an extensive testing protocol, including simulated and real-world scenarios, to identify and rectify algorithmic inaccuracies.	Establish a rapid-response team to address emerging issues promptly, providing over-the-air updates to users.
Security Measures:	Employ encryption technologies, secure data transmission protocols, and regular security audits.	Activate a response team to isolate and address security breaches, inform affected users, and implement corrective measures.
Adaptive Integration Framework	Develop a flexible integration framework to accommodate varying vehicle systems and platforms.	Maintain an ongoing collaboration with manufacturers, ensuring swift adaptations to changes in vehicle technology.
User Engagement and Education	Implement a comprehensive user engagement plan, including tutorials, FAQs, and responsive customer support.	Monitor user adoption rates and feedback, adjusting educational materials and support systems as needed.
Regulatory Monitoring	Establish a dedicated team to monitor and interpret changes in road safety regulations.	Develop a rapid-response plan to adapt the Virtual Assistant to meet new legal requirements and maintain its legal operation.

4.8 Criteria of acceptance:

Conditions that Must be Met for the Project to be Considered Successful:

Name	Condition	Rationale
Hazard Detection Accuracy:	The Virtual Assistant must achieve a minimum accuracy rate of 95% in identifying and alerting users to hazardous road conditions.	Ensuring a high level of accuracy is critical for the reliability and effectiveness of the system.
Real-time Responsiveness	The system must provide real-time hazard alerts with a latency of no more than two seconds from the detection of a hazard.	Timely alerts are crucial for enabling users to react promptly to potential dangers.
User Adoption Rate	The Virtual Assistant must achieve a user adoption rate of at least 80% within the first six months of deployment	User adoption is indicative of the system's perceived value and ease of use.
Compliance with Data Privacy Regulations	The system must adhere to all relevant data privacy regulations and standards, ensuring the secure handling of user data.	Protecting user privacy is a fundamental aspect of the project's success and ethical considerations.
Integration with Vehicle Systems:	The Virtual Assistant must seamlessly integrate with a minimum of 90% of commonly used vehicle systems and platforms.	Compatibility with diverse vehicle models enhances the system's applicability and market reach.

4.8.1 Quality Criteria and Tests:

- Algorithmic Precision and Recall:
  - Test: Conduct precision and recall tests using diverse datasets to evaluate the algorithm's accuracy in identifying hazards without generating false positives.
- Latency Testing:
  - Test: Simulate various road conditions and scenarios to assess the system's real-time responsiveness, ensuring it meets the specified latency criteria.

- Usability Testing:
    - Test: Engage a representative sample of users in usability testing to evaluate the clarity, intuitiveness, and overall user experience of the interface.
- Security Audits:
    - Test: Conduct regular security audits to identify and address potential vulnerabilities, ensuring the system complies with data privacy and security regulations.
- Integration Compatibility Testing:
    - Test: Verify the Virtual Assistant's compatibility with a range of vehicle systems through integration tests, ensuring seamless communication and functionality.
- User Feedback Analysis:
    - Test: Analyze user feedback regarding hazard accuracy and system performance to identify areas for improvement and refinement.
- Regulatory Compliance Checks:
    - Test: Regularly review and update the system to ensure compliance with evolving road safety regulations and legal requirements.

4.9 Human resources

List of Roles and Responsibilities:

Rol	Responsibilities
Project Manager	Oversee the overall planning, execution, and delivery of the Virtual Assistant project.
	Define project scope, goals, and deliverables
	Coordinate with stakeholders, ensuring alignment with project objectives.
	Monitor and report project progress, risks, and resource utilization.
Lead Developer:	Lead the software development team in implementing algorithms and system functionalities.
	Oversee code quality, ensuring adherence to coding standards and best practices.
	Collaborate with other teams to integrate software components seamlessly.

User Interface (UI) Designer:	Design intuitive and visually appealing user interfaces for both mobile and in-vehicle applications.
	Ensure the UI enhances user experience and aligns with usability principles.
	Collaborate with developers to implement UI designs effectively.
Machine Learning Engineer:	Develop and implement machine learning models for hazard detection and system adaptation.
	Continuously improve algorithms based on user feedback and evolving road conditions.
	Collaborate with the development team to integrate machine learning components.
Backend Developer	Design and implement the backend infrastructure for data processing, storage, and retrieval
	Develop APIs for seamless communication between frontend and backend components.
	Ensure the scalability and efficiency of the backend architecture.
Data Security Specialist	Implement and maintain robust data security measures to protect user information.
	Conduct regular security audits and address potential vulnerabilities.
	Ensure compliance with data privacy regulations.
User Engagement Specialist:	Develop and execute user engagement strategies, including tutorials and onboarding materials
	Gather user feedback and insights to enhance user experience.
	Provide responsive customer support to address user queries and concerns.

Name	Rol	Cost
Rodrigo Resendiz	Project manager	$4000
Kevin Alvarez	Lead Developer	$3000
Omar Bravo	UI Designer	$2500
Peter Anguila	Backend Developer	$1200
Benny Urdaneta	Backend Developer	$1200
Matew Ramirez	Machine Learning Engineer	$2000
Jaime Resses	User Engagement specialist	$2000
Andres Lopez	Data Security specialist	$2000
Claudia Galvez	UI Designer	$2000
Andrea Doriga	Backend Developer	$2000

4.10 Budget

Detail of Estimated Costs:

Scales	Rol	Cost
Sr	Project manager	$4000
	Lead Developer	$3000
	UI Designer	$2500
	Backend Developer	$2000
Jr	Backend Developer	$1200
Sr	Machine Learning Engineer	$2000
	User Engagement specialist	$2000
	Data Security specialist	$2000
Development Tools and Licenses		$1000
Testing Team Salaries		$4000
Testing Tools and Licenses		$1000
Simulation and Test Environments		$2000
Infrastructure and Cloud Services:	Backend Infrastructure	$3000
	Cloud Services (AWS/Azure):	$1500
Personnel Training and Development:	Training Programs	$1500
	Skill Enhancement Workshops	$500
Regulatory Compliance and Certification:	Legal Consultation for Compliance	$1500
	Certification Fees	$800
User Engagement and Support	User Feedback Analysis Tools	$1200
	Customer Support Platform	$1200
Contingency and Unforeseen Expenses	Contingency Fund (10% of Total Budget)	10% of total budget

3.5 Change Control Document

In the dynamic landscape of software development, the Change Control Document plays a pivotal role in managing alterations to the original project scope, ensuring that modifications are implemented in a systematic and controlled manner. This document serves as a structured framework to evaluate, authorize, and track changes, thereby safeguarding the project's integrity and aligning with the overarching objectives.

As we embark on the development journey of the Virtual Assistant for Hazardous Road Recognition, the Change Control Document becomes an indispensable tool for maintaining transparency, minimizing risks, and fostering adaptability in response to evolving project needs. Through this document, we establish a robust mechanism for change management, emphasizing the importance of thorough analysis, stakeholder collaboration, and strategic decision-making to propel the project towards successful fruition.

UNIVERSIDAD DE GUADALAJARA

Centro Universitario de los Valles

Maestría en Ingeniería de Software

Software Configuration Management Project

Change request and policies Virtual Assistant for Recognizing Dangerous Roads.

Professor:
Dr. Omar Ali Zataraín Durán
Student:
Rodrigo Reséndiz Ávila 222977423

Master in software engineering
Nov 17, 2023
Ameca, Jalisco

1.Control version

Table 1.1 Policies for accepted any change request

Date	17/11/2023	Document	Committee board
Configuration Item Description	Policies	Initial release	
Documented on the repository of the project			Yes
Previous document version		None	
Comments			
No Comments			

Table 1.2 first version of the change request control

Date	06/09/2023	Document	Configuration item
Configuration Item Description	CR	Initial release	
Documented on the repository of the project			Yes
Previous document version		None	
Comments			
No Comments			

Table 1.3 update version for change request and policies

Date	28/11/2023	Document	Change_Reques_V1.1
Configuration Item Description	Tables are added to make the policies as clear as possible, as well as the scales for making decisions on change requests.		
Documented on the repository of the project			Yes
Previous document version		Configuration item, Committee board	
Comments			
No Comments			

2. Introduction

This document delineates change requests submitted by the client, accompanied by a comprehensive analysis to facilitate policy execution and informed decision-making for acceptance determination.

3. Policies

To make any change to the established baseline, acceptance of the following policies is necessary.

1.Any change request outside of what is stipulated in the baseline will require a maximum of 1 week to be analyzed.
2. Accepted change requests must have time and budget that will be added to what is already stipulated in the baseline.
3. Change requests requiring new staff will have a low priority.
4. Change requests that affect more than 3 modules will not be approved.
5. Establish a review committee that includes representatives from different functional areas to ensure a comprehensive evaluation of change requests.
6. Require detailed documentation for each change request, including the reason, expected benefits, and any potential impacts on the project scope, time, and budget.
7. Establish a tracking system that allows stakeholders to get regular updates on the status of change requests, including milestones and key results.
8. Each change request shall include a detailed assessment of potential associated risks and challenges, as well as mitigation strategies.
9. All change requests will be prioritized based on their impact on project objectives, urgency, and alignment with the overall strategy.
10. If the risk is greater than 5-6 points, the change request is rejected.

3.1 Process for approved Change Request.

1. Any team member or stakeholder submits a Change Request (CR) containing details about the proposed change.
2. A designated responsible party reviews the CR to determine its relevance and initial feasibility.
3. If the RFC passes the initial assessment, the development team conducts a comprehensive technical analysis, including effort estimations, risk identification, and assessment of the impact on the project.
4. The team prioritizes the RFC based on its importance, impact, and urgency.
5. The team, along with the Scrum Master or team leader, makes the final decision to approve the RFC.
6. The approval of the RFC is communicated to all stakeholders, and the implementation of the change is planned.
7. The change is implemented in accordance with the team's development and version control practices.
8. Follow-up is conducted after implementation to assess the impact and whether the objectives are met.

3.2 Process for Not approved Change Request

1. Any team member or stakeholder submits a Change Request containing details about the proposed change
2. A designated responsible party reviews the RFC to determine its relevance and initial feasibility.
3. If the RFC does not pass the initial assessment or the technical analysis, reasons for denial are documented.
4. The decision to not approve the RFC is communicated to the requester, with clear explanations of the reasons behind the denial.
5. All non-approved RFCs, along with their denial reasons, are archived for future reference
6. The requester is informed that the RFC may be reviewed in the future if circumstances change or if additional information is provided.

4. Scale for measure Risk

Table 4.1 Scales for measure risk

Budget	**Time**	**Human resources**	**Affected Modules**	**Work Effort**	**Risk**
10%	1 to 3 Months	0	1	10%	Low (1)
10% to 15%	3 to 5 months	1 to 2	2 to 3	10% to 20%	Medium (2)
>15%	>5 months	>2	>= 3	>20%	High (3)

The following table will discuss how work effort is measured based on the affected modules, which in turn will be classified as new modules or modifications to those already established.

Table 4.2 Measure for work effort

Work effort		**%**
New module	1	15%
	2 to 3	25%
Modified	1	5 %
	2 to 3	10 %

5. Change request CR-01

Table 5.1 CR-01 Specifications and introduction

Change Request ID:	CR-01	Requestor	Omar Ali Zatarain Durán
Date	08/09/2023	Project manager	Rodrigo Resendiz Avila
Describe the Requested Change			
Incorporate criminal behavior history on the road; this includes the following: • Hit and run cases by road, GPS location, year, daytime. • Shootings between gangs and solo shooters • Murders and mass murders. • Kidnapping • The system is expected to be connected to a criminal records database to accomplish the criteria above.			

5.2 Affected Modules

- User Interface (1)
- Integration with criminal history database (1)

5.3 New modules

- Criminal data base (3)
- Criminal Behavior Recording (3)

Table 5.2 CR-01 SWOT Matrix specification.

SWOT	
STRENGTHS (+)	WEAKNESSES (-)
The System will contain detailed information on the vehicle history, to provide a correct response to those traveling on the road.	The system will have a very robust database that would be incapable of processing data in the shortest possible time. The antecedents may vary according to the power that dictates them, which causes more processing time.
OPPORTUNITIES (+)	THREATS (-)
- Attract government investment so that the system can support the robustness of the database.	It completely deviates from the project proposed in the baseline. This would affect the requirements raised.

Table 5.3 budget impact

Budget impact				
Money	Time	H.R	Work effort	Affected modules
$ 7,200	1 Month	4	High	4

The work effort is high due to the affected modules and the new modules, which, as mentioned in table 4.2, obtain a high percentage and according to table 4.1 a high result is obtained.

5.4 Conclusion for change request CR-01

Because the change request requires too many modifications from the baseline, and takes a high percentage of the budget ($ 7,200), the decision is made to reject the request and leave it for future updates.

6. Change request CR-02

Table 6.1 CR-02 Specifications and introduction

Change Request ID:	CR-02	Requestor	Omar Ali Zatarain Durán
Date	06/10/2023	Project manager	Rodrigo Resendiz Avila
Describe the Requested Change			
The client requests to include an alarm sent to the police when the user encounters a risky situation while on the street.			

6.1 Affected Modules

This change request is not affected by any module, since in the baseline it is present in module 7 “Notifications to Authorities and Emergency Services”, so we will only talk about it in more detail.

6.2 SWOT Matrix

Table 6.2 CR-02 SOWT Matrix

SWOT	
STRENGTHS (+)	WEAKNESSES (-)
Having direct interaction with emergency services could help reduce the rate of incidents on the road, in turn alerting users about what is happening, the season in which it occurs and the times in which they occur most frequently.	Possible Misuse: There is a risk that the alarm function may be misused, generating false alarms. User Privacy: Implementing an alarm function must carefully manage user privacy to avoid potential concerns.
OPPORTUNITIES (+)	THREATS (-)
Increased User Trust: Improving security could increase user trust in the virtual assistant. Collaboration with Authorities: The alarm feature could open opportunities for collaborations with local authorities for a more effective response. Market Expansion: The inclusion of robust security features could attract new users concerned about their security.	Legal Concerns: The alarm feature could face legal and regulatory challenges related to privacy and the transmission of information to authorities. Negative Perception: If the feature is not implemented correctly, it could result in a negative perception of the virtual assistant as intrusive or excessive.

6.3 Conclusion for change request CR-02

Because this requirement is present in the baseline, it is necessary to inform the client who requests it, to clarify doubts and continue if any other changes are necessary. Therefore, this request is on hold.

7 Change request CR-03

Table 7.1 CR-03 Specifications and introduction

Change Request ID:	CR-03	Requestor	Omar Ali Zatarain Durán
Date	06/10/2023	Project manager	Rodrigo Resendiz Avila
Describe the Requested Change			
The client requested that the cell phone camera and microphone be activated, recorded, and uploaded to the cloud when in a risky situation.			

7.1 Affected modules

- Monitoring and Maintenance (1)

7.2 New modules

- Recording and Cloud Upload Module (2)
- Cloud Server (2)

7.3 SWOT Matrix

Table 7.2 CR-03 SOWT Matrix

SWOT	
STRENGTHS (+)	WEAKNESSES (-)
Improved Safety: Activating the camera and microphone in risky situations can improve the ability to collect relevant information. Real-Time Evidence: Real-time recording provides valuable evidence to evaluate risk situations and make informed decisions. Quick Response: The ability to upload data to the cloud can allow for a faster response from authorities or emergency services.	Privacy Concerns: Activation of the camera and microphone raises privacy concerns, and their management must be careful to avoid negative perceptions. Possible Misuse: There is a risk that the function will be misused or interpreted as invasive.
OPPORTUNITIES (+)	THREATS (-)
Collaboration with Authorities: The function can be a basis for collaborations with authorities to improve citizen security. User Trust: Secure and ethical implementation can increase user trust by providing an additional level of security.	Legal and Regulatory Challenges: The function may face legal and regulatory challenges related to privacy and the handling of sensitive data. Negative User Perception: Incorrect implementation or insufficient privacy management can lead to negative user perception.

Table 7.3 budget impact

Budget impact				
Money	Time	H.R	Work effort	Affected modules
$ 3,500 (12%) (2)	1 Month (1)	0	Medium	1

7.4 Conclusion for change request CR-03

After the analysis carried out for this requirement and finding that the impact is not too serious, and falls within the budget established for the change requests. Therefore, it is approved.

8 Change request CR-04

Table 8.1 CR-03 Specifications and introduction

Change Request ID:	CR-04	Requestor	Omar Ali Zatarain Durán
Date	06/10/2023	Project manager	Rodrigo Resendiz Avila
Describe the Requested Change			
Due to government regulation, the system should transmit all the information encrypted.			

8.1 Affected modules

- Notifications and Alerts (1)

8.2 New modules

- Data Encryption (2)
- Key Management

8.3 SWOT Matrix

Table 8.2 CR-04 SOWT Matrix

SWOT	
STRENGTHS (+)	WEAKNESSES (-)
Regulatory Compliance: Implementation of encrypted transmission ensures compliance with government regulations. Improved Security: Encryption strengthens the security of information, protecting it against possible threats and unauthorized access. Improved User Confidence: Strengthened security can increase user confidence by ensuring the protection of their data.	Additional Resource Requirements: Implementing encryption may require additional resources, both in terms of time and computational capacity. Possible Performance Impact: Encryption can have an impact on system performance, and this should be managed carefully.

OPPORTUNITIES (+)	THREATS (-)
Competitive Differentiation: Complying with government regulations can differentiate the system as safe and reliable compared to the competition. Possibility of Government Collaboration: Adherence to regulations may open opportunities for government collaborations or strategic partnerships.	Changes in Regulations: Changes in government regulations may require frequent adjustments to the encryption implementation. Implementation Challenges: Improper implementation of encryption could result in security vulnerabilities, defeating the initial purpose.

Table 8.3 budget impact

Budget impact				
Money	Time	H.R	Work effort	Affected modules
$ 4,000 (16%) (2)	2 Month (1)	1	Medium	1

8.4 Conclusion for change request CR-04

For this change request, the risk is medium, due to human resources considerations and the work effort marked in points 8.1 and 8.2, even with the consideration of medium risk, this request must be approved since the requirement is necessary to comply with the regularizations set by the government, therefore, are approved.

3.6 Status accounting

This section provides a detailed description of the CRs and the way in which they were implemented in order to know if this CR was implemented correctly. In the case of the initial version of the document, it has the description of the implementation of the accepted CRs. Part of the important information for its subsequent referencing in the documentation is the RC information, the problems presented over time, the money used, the time used and the human resources that were required for the implementation of the RC and the result obtained of the implementation. The information shown in the state accounting document will contain the following points.

- **Approval Date**: The date the CR was approved.
- **Implementation Date:** The date the CR was implemented in the project.
- **CR ID:** The identification for the change request.
- **CR description**: Description of the CRs.
- **Affected CI**: Elements affected by the implementation of the CR.
- **Estimated budget**: Expected budget for the development of the CR
- **Estimated time:** Estimated time for the implementation of the CRs.
- **Estimated people:** Personnel assigned or used for development.
- **Realized budget**: Money finally used to implement the CR
- **Time spent**: The actual time in days required to implement the CR.
- **People Involved**: The name of the people who were involved during the CR implementation.
- **Human resource performance**: The way in which the staff performed.
- **Result:** Conclusion of the CR

UNIVERSIDAD DE GUADALAJARA

Centro Universitario de los Valles

Maestría en Ingeniería de Software

Software Configuration Management Project

Status accounting for Virtual Assistant for Recognizing Dangerous Roads.

Professor:
Dr. Omar Ali Zataraín Durán
Student:
Rodrigo Reséndiz Ávila 222977423

Master in software engineering
Nov 24, 2023
Ameca, Jalisco

Table 3.6.1 First version for status accounting

Date	28/11/2023	Document	StatusAccounting V0.1
Configuration Item Description	Initial sample of status accounting, breakdown of expenses and time.		
Documented on the repository of the project			Yes
Previous document version	None		
Comments			
No Comments			

Table 3.6.2 Update Status Accounting

Date	28/11/2023	Document	StatusAccounting V1.0
Configuration Item Description	Improving the presentation with tables to better represent who the tasks were assigned to and how much the estimated budget was and what was actually spent.		
Documented on the repository of the project			Yes
Previous document version	StatusAccounting V0.1		
Comments			
No Comments			

Introduction

In this document, there is a control of the changes, seeking to evaluate how many resources were used to complete the development of the change and in this way be able to compare them with the resources that were initially estimated to be used. This document also allows us to briefly detail which members of the team involved participated in the change and how they dealt with the change.

Table 3.6.3 Status accounting report for the implementation of CR-03

Approval date	27/10/2023	Date of implementation	27/11/2023
identifier CR	CR-03		
CR description			
The client requested that the cell phone camera and microphone be activated, recorded, and uploaded to the cloud when in a risky situation.			
IC affected			
Monitoring and Maintenance			
Estimated budget	$ 3,500 USD		
Estimated time	1 Month		
Planned Human Resource	1 Sr Developer, 1 Jr Developer		
budget used	$ 2,700 USD	Difference	$ 800 USD

Time spent	1 Month	Difference	None
Personnel involved			
• Peter Anguila			
Human resource performance			
Since the request only required a small change in the module, the work team assigned the task to the Jr developer, removing a burden and monetary value from the Sr developer.			
Result			
The application performs a correct synchronization with the data to the cloud, and performs the video and voice recording requested by the client.			

Table 3.6.4 Status accounting report for the implementation of CR-04

Approval date	27/10/2023	Date of implementation	27/12/2023
identifier CR	CR-04		
CR description			
Due to government regulation, the system should transmit all the information encrypted.			
IC affected			
Notifications and Alerts			
Estimated budget	$ 4,000 USD		
Estimated time	2 Month		
Planned Human Resource	1 Sr Developer		
budget used	$ 4,000 USD	Difference	$ 0 USD
Time spent	2 Month	Difference	None
Personnel involved			
• Andres Lopez			
Human resource performance			
The team needed to take a course to learn about the established government regulations, and know how to implement them in the project so that it complies with what was requested.			
Result			
The System has the regulations established by the government, as well as privacy between users and authorities.			

3.7 Audit

This section of the work is the final part of the SCM process made to Virtual Assistant for Recognizing Dangerous Roads. This is a document in which the quality of a product is verified after its processing and development. This section seeks to examine if the practices carried out by the development team were the most correct, if the documentation was carried out correctly, if the communication and communication channels were adequate to express any doubt or communicate information between departments, At the same time, it is also evaluated if the treatment within the work groups was cordial and encouraged a pleasant work environment. In addition to communication, the audit seeks to assess whether all change request were completed. For the writing of this document, the following points are considered:

- CR accepted
- Risks
- Team and staff performance
- Resource utilization
- Methodology
- Communication channels
- Standards and Help

UNIVERSIDAD DE GUADALAJARA

Centro Universitario de los Valles

Maestría en Ingeniería de Software

Software Configuration Management Project

Audit for Virtual Assistant for Recognizing Dangerous Roads.

Professor:
Dr. Omar Ali Zataraín Durán
Student:
Rodrigo Reséndiz Ávila 222977423

Master in software engineering
Nov 28, 2023
Ameca, Jalisco

Table 3.7.1 Initial audit

Date	28/11/2023	Document	Audit V1.0
Configuration Item Description	Initial audit for the proyect.		
Documented on the repository of the project			Yes
Previous document version		none	
Comments			
No Comments			

Points Considered:

Change Requests Accepted (CR):

- Verify the status of all change requests, ensuring they were accurately documented and implemented.

Risk Evaluation:

- Assess the identified risks associated with real-time data transmission, privacy concerns, and potential misuse. Evaluate how effectively they were managed and mitigated throughout the development process.

Team and Staff Performance:

- Examine the performance of the development team and staff involved in implementing security features, considering factors such as productivity, collaboration, and adherence to timelines.

Resource Utilization:

- Evaluate the utilization of resources, including time, personnel, and technology, to ensure efficiency and optimal allocation in the development of security-related functionalities.

Methodology Review:

- Scrutinize the adopted development methodology to ensure adherence to best practices and its effectiveness in achieving project goals.

Communication Channels:

- Assess the effectiveness of communication channels within and between departments, particularly focusing on features like in-app communication and notification systems.

Standards and Help:

- Review adherence to established standards and the availability of supportive resources, ensuring compliance with industry norms and best practices.

Audit Objectives:

- Verify the correctness of development practices for security-related features.
- Confirm the accuracy of documentation procedures for security implementations.

- Evaluate the adequacy of communication channels for expressing doubts or conveying information regarding security features.
- Assess the overall workgroup environment for cordiality and a positive atmosphere during the implementation of security functionalities.
- Confirm the completion status of all change requests related to security enhancements.

This audit report aims to provide a comprehensive overview of the SCM process's effectiveness in ensuring the quality and integrity of the Virtual Assistant for Hazardous Road Recognition project. Through thorough examination of the aforementioned points, we strive to ensure that the development and processing phases align with industry standards and project security objectives.

References

[1] F. J. Buckley, «Implementing a Software Configuration Management Environment,» Computing practices, pp. 56-61, 1994.

[2] "IEEE Recommended Practice for Software Design Descriptions," in IEEE Std 1016-1998 , vol.,no., pp.1-23, 4 Dec. 1998, doi:10.1109/IEEESTD.1998.88828.

[3]Romero, R. SIPaF v1.1. GitHub. Retrieved December 2, 2022, from https://github.com/3xR-L/SCM_2022_RomeroRaul/blob/main/Baselines/SIPaF%20v1.1.pdf

[4] Resendiz,R. BaseLine V1.1. Github. Retrived November 28, 2023 from https://github.com/RodrigoResendiz/Software--configuration-management/blob/main/BaseLine_RodrigoResendiz.pdf