Universidad de Guadalajara
Centro Universitario de los Valles
Maestría en Ingeniería de Software

Software Configuration Management Project

Author: Alejandro Prado Lopez

Professor: Phd. Omar Ali Zatarain Durán

Fecha: 29 de Noviembre de 2024

1. Introduction

This document outlines the Software Configuration Management (SCM) strategy for the development and maintenance of a web application aimed at assisting healthcare professionals in managing and monitoring the rehabilitation of patients with facial paralysis.

The application supports healthcare professionals in their work by enabling the creation and management of patient profiles, the design and assignment of personalized rehabilitation routines, and the tracking of patient progress. Additionally, it offers a dashboard for visualizing progress and key patient information.

The following sections provide detailed information about the SCM process, including baselines, configuration identification, control, status accounting, and auditing.

2. Software Configuration Management

Software Configuration Management (SCM) is a systematic process for managing changes in software products throughout their lifecycle. It ensures that software systems are consistent, traceable, and well-documented by controlling their configurations. SCM enhances team collaboration, minimizes integration risks, and ensures compliance with quality standards by keeping track of all software changes.

2.1. Configuration Identification

Establishing a baseline by identifying all components of the software, such as source code, documentation, libraries, and dependencies. This helps in managing and versioning these elements effectively.

2.2. Configuration Control

Ensuring that any changes to the software baseline are reviewed, approved, and documented. This process prevents unauthorized modifications and maintains stability in the development process.

2.3. Configuration Status Accounting

Recording and reporting information about the current state of the software configuration, including version numbers, change history, and build status. This provides visibility into the project’s progress and integrity.

2.4. Configuration Auditing

Verifying that the software complies with its specifications and that all changes have been properly implemented and documented. Audits ensure quality and traceability in the software development process.

3. Baseline - Configuration Identification

The baseline defines the foundation of the project and serves as a reference point for tracking changes, ensuring alignment with the project's goals and requirements.

3.1. Product description

The web application is designed to assist healthcare professionals in managing and monitoring the rehabilitation of patients with facial paralysis. The platform enables the creation of patient profiles, where professionals can input and track patient information.
It allows the development of personalized routines for each patient's needs.

Main features:

- **Patient Profiles:** Creation and management of profiles. Storing relevant information like routine's duration time, patient' progress, patient's images, professional's notes, and all related information.
- **Custom Routine Creation:** Design of routines focused on each patient's needs.
- **Routine Assignment:** The professional can assign a specific routine to a patient.
- **Progress Tracking:** Through the information got from the routines the professional can track the progress of the patient in each session.
- **Dashboard:** The professional has a view of each patient's progress and the main focused information.

This software is just a part of a complex system that includes a mobile application and cloud API.

3.2. Functional Requirements

- **Authentication**
  - Log in into the application
- **Patient Profile Management**
  - The system must allow professionals to create and manage patient profiles.
  - Each patient profile should store relevant information such as routine's duration time, personal details, progress reports, images, and professional's notes.
- **Custom Routine Creation**
  - Healthcare professionals must be able to design and create custom rehabilitation routines based on the patient's specific needs.

    - The system must allow professionals to update and modify routines as needed.
- **Routine Assignment**
    - The system should allow professionals to assign specific rehabilitation routines to individual patients.
- **Progress Tracking**
    - The system must track the patient's progress by storing and analyzing data from each rehabilitation session.
    - Progress reports should be generated based on patient activities, and these reports should be accessible by professionals through the web application.
- **Dashboard**
    - A dashboard must be available for healthcare professionals, providing an overview of patient profiles, progress reports, and other key information in a user-friendly interface.
    - The dashboard must display high-level patient progress summaries and detailed individual reports.

3.3. Non-Functional Requirements

- **Security**
    - Data encryption must be used for both data at rest and data in transit.
    - The system must implement user authentication and role-based access control to restrict access to sensitive information.
- **Scalability**
    - The system must be able to handle increasing amounts of patient data.
- **Performance**
    - The system must respond to user interactions (e.g., loading patient profiles, assigning routines, and updating progress) within an acceptable time frame (less than 4 seconds for most actions).
    - Data retrieval and processing should be optimized to ensure smooth operation for healthcare professionals.
- **Usability**
    - The system interface must be intuitive and easy to use for healthcare professionals, ensuring minimal training is required.
    - Information should be presented clearly, with key patient information and progress easily accessible from the dashboard.
- **Reliability and Availability**
    - The system must be available 99.9% of the time, ensuring minimal downtime for healthcare professionals using the platform.
    - Backup mechanisms should be in place to prevent data loss in case of system failure.
- **Compatibility**
    - The web application must be compatible with all major web browsers

- **Maintainability**
  - The architecture should support easy integration with future modules or updates in the mobile application or cloud API.
- **Data Consistency and Integrity**
  - The system must ensure that data is consistently updated across all platforms (web, mobile, cloud) to avoid any discrepancies in patient progress and routines.

3.4. System Modules

The project is divided into the following modules, with tasks and their corresponding time estimations:

3.4.1. Authentication

Task	Estimation
Manage Users	5 days
Log In and Log Out	3 days
Recover Password	2 days
Authentication by JWT token for mobile devices	10 days

3.4.2. Patient Profile Management

Task	Estimation
Create patient	2 days
Update routine's duration time, personal details, progress reports, images, and professional's notes by API	5 days
Show patient information	2 days

3.4.3. Custom Routine Management

Task	Estimation
Create routines with exercises	3 days

Update exercises in routines	3 days
Show routine information	2 days

3.4.4. Routine Assignment

Task	Estimation
Assignment routine	3 days
Show routine information	3 days

3.4.5. Progress Tracking

Task	Estimation
Storing session data from device	5 days
Generate page to display session information	8 days
Add graphics to tracking page	5 days

3.4.6. Dashboard

Task	Estimation
summary of patients' information	8 days
filters in dashboard configuration	5 days

3.4.7. Configuration

Task	Estimation
Set up of the project(servers, github, CI/CD, Docker)	20 days
Configuration and deploy to production	5 Days

3.4.8. Management

Task	Estimation
Project Manager	35 days
Testing	30 days

3.5. Project Costs

The cost structure for the project is based on hourly billing rates for different roles involved in the software development lifecycle. These rates reflect the specialized expertise and effort required for each phase of the project, ensuring the delivery of a high-quality product.

Process	Description	Hourly rate
Development	Back-end and Front-end implementation	$150.00
Management	Project planning, coordination, ensuring timely delivery	$200.00
Testing	Testing and debugging to ensure the product is robust, secure and meets all the requirements	$100.00

The project's overall cost is calculated based on the estimated duration, daily rates for each activity, and the associated efforts for development, configuration, testing, and management. A time buffer is included to account for unforeseen challenges and ensure timely delivery.

Description	Days	Weeks	Cost per day	Total
Development	74	14.8	$1,200.00	$88,800.00
Configuration	25	5	$1,200.00	$30,000.00
Testing	30	6	$800.00	$24,000.00

Management	35	7	$1,600.00	$56,000.00
Time Buffer	32	6.4	$1,200.00	$38,400.00
Total				**$237,200.00**

3.6. Team

The project team consists of highly skilled professionals, each contributing to different aspects of the software development process. Their monthly salaries reflect their expertise and roles in the project.

- **Junior Developer:** Responsible for frontend implementation and assisting with routine backend tasks.
    - Salary: **$12,000/month**
- **Mid-Level Developer:** Focused on backend development, API integration, and database management.
    - Salary: **$25,000/month**
- **Project Manager:** Oversees the project, ensuring smooth communication between stakeholders and team members, while tracking progress and resolving challenges.
    - Salary: **$28,000/month**
- **Tester:** Conducts quality assurance activities to ensure the software meets functional and non-functional requirements.
    - Salary: **$15,000/month**

NOTE: A month has approximately 4.35 weeks.

3.7. Internal Estimation

The project's timeline is divided into key phases: configuration, development, testing, and management. Each phase has an estimated duration based on the scope of tasks and team allocation.

Description	Role	Weeks	Total
Configuration	Mid Developer	5	$28,736.00
Development	Mid Developer	7	$40,240.00
	Junior Developer	10	$27,587.00

Testing	Tester	6	$20,690.00
Management	Project Manager	7	$45,058.00
Total internal Cost			**$162,311.00**

The project is expected to take approximately **27.4 weeks** (including the time buffer), which is broken down as follows:

- **Core Phases:** 21 weeks
- **Time Buffer:** 6.4 weeks

4. Configuration Control

This section describes the four change requests received during the project. Each request is analyzed in detail, including its requirements, affected modules, estimated costs, potential revenue, and a SWOT analysis.

The Change Control Board (CCB) evaluates these requests based on predefined criteria, and voting determines whether the changes are approved, deferred, or rejected.

4.1. Change Request 1 - Multi-location Clinic

Key: CR001

Description: The client wants to add a new feature to the system. Since the client has many clinics in Jalisco. So, the client wants to differentiate the sessions and patients from each clinic.

Requirements:

- Each Clinic has its patients and sessions.
- Professionals could attend only patients from the clinic assigned.

Modules affected:

- **Authentication**: Professional will see only the information based on the clinic assigned.
- **Patient Profile Management:** Patient will be assigned only one clinic
- **Dashboard**: Display information only for the clinic assigned to the professional
- **Progress Tracking**: Device has to know which clinic is set for it

Tasks:

Task	Estimation
Add clinic relationship with patients and sessions	3 days
Assign professionals to a clinic	5 days
Redirect professionals based on the clinic assigned	4 days
Display information on dashboard only for the professional's clinic	5 days
Add clinic to API requests	10 days

Feature costs:

Description	Days	Weeks	Cost per day	Total
Development	27	5.4	$1,200.00	$32,400.00
Testing	12	2.4	$800.00	$9,600.00
Management	10	2	$1,600.00	$22,400.00
Time Buffer	7	1.4	$1,200.00	$8,400.00
Total				**$72,800.00**

Internal cost:

Description	Weeks	Total
Development		
Mid Developer	2	$11,500.00
Junior Developer	4	$11,000.00
Testing		
Tester	2.4	$8,300.00
Management		
Project Manager	2	$12,900.00
Total internal Cost		**$43,700.00**

Potential revenue:
Estimate potential revenue generated by this feature is **$29,100**.

SWOT Matrix:

- Strengths
  - Aligns with the client's operational requirements, ensuring the system addresses their needs effectively.
- Weaknesses
  - Adds workload amidst other concurrent development tasks, increasing team pressure.
- Opportunities
  - Positions the platform as a scalable solution for healthcare providers with multiple locations, enhancing market appeal.
- Threats
  - Potential delays in the project timeline due to the complexity of changes and testing

CCB Criteria and Voting:

- CEO - Supports the Change Request
    - **Reason**: This feature is critical to meet the client's requirements and maintain their satisfaction. It also aligns with the company's goals of offering scalable solutions to multi-location healthcare providers.
- Financial Department - Supports the Change Request
    - **Reason**: While the potential revenue of $29,100 does not completely offset the cost of $72,800, the feature enhances long-term client retention and opens opportunities for future upsells.
- Human Resources - Supports the Change Request
    - **Reason**: The feature aligns with the team's capabilities, but HR advises close monitoring of workloads given the simultaneous demands of other ongoing projects.
- Project Manager - Supports the Change Request
    - **Reason**: The Project Manager supports the change, as it is feasible within the allocated buffer time and ensures the client's operational goals are met. Additional resources may be required for testing to avoid delays in core deliverables.

Final Outcome: With unanimous support, the recommendation is to proceed with the change request.

4.2. Change Request 2 - Other paralysis support

Key: CR002

Description: The client requests new functionalities to include new types of paralysis, such as arm and leg paralysis.

Requirements:

- Support new routines for arm paralysis.
- Display information about arm paralysis tracking progress on the dashboard.
- Support new routines for leg paralysis.
- Display information about leg paralysis tracking progress on the dashboard.

Modules affected:

- **Patient's profile:** Profile needs to have extra information related to the leg and arm paralysis.
- **Custom Routine Creation:** Routines need to have new animations and add new information related to new exercises.
- **Progress Tracking:** Have a new section to display information about leg and arm paralysis.
- **Dashboard:** Have a new section to display important information related to leg and arm paralysis.

Tasks:

Task	Estimation
Add new attributes to the database and set on models for patients.	8 days
Create new forms to create routines for arm and leg paralysis.	10 days
Create progress tracking sections for leg and arm paralysis.	8 days
Display eg and arm paralysis information on the dashboard.	9 days
Support new information for leg and arm paralysis on the API	15 days

Feature costs:

Description	Days	Weeks	Cost per day	Total
Development	50	10	$1,200.00	$60,000.00
Testing	23	4.6	$800.00	$18,400.00
Management	15	3	$1,600.00	$24,000.00
Time Buffer	10	2	$1,200.00	$12,000.00
Total				**$114,400.00**

Internal cost:

Description	Weeks	Cost	Total
Development			
Mid Developer	4	$5753.42	$23,013.68
Junior Developer	7	$2,761.64	$19,331.48
Testing			
Tester	3	$3452.05	$10,356.15
Management			
Project Manager	3	$6443.83	$19,331.49
Total internal Cost			**$72,032.8**

Potential revenue:
Estimated potential revenue generated by this feature is **$42,367.00**. This means a 37% profit.

SWOT Matrix:

- Strengths
  - Team has a lot of experience working together.
  - Feature does not affect the core of the application.
- Weaknesses
  - There are two projects coming.
  - The Mid developer will be moved into another project.
- Opportunities
  - The client has more ideas that wants to apply to the application.

    - The Junior developer can learn from the process to add new features.
- Threats
    - Tester will leave the company next month.

CCB Criteria and Voting:

- CEO - Supports the Change Request:
    - **Reason**: The projected revenue from the client ($114,400) exceeds the internal costs ($72,032.80), yielding a profitable margin. Since the feature aligns with the company's strategic vision of creating comprehensive rehabilitation solutions
- Financial Department - Supports the Change Request:
    - **Reason**: The feature cost ($114,400) provides a profitable margin over the internal costs, promising a strong return. The department sees this feature as financially sound, despite upcoming projects, as it doesn't require additional external funding.
- Human Resources - Does not support the Change Request:
    - **Reason:** Lack of resources, There is no way to get a new tester and mid developer in a short period of time.
- Project Manager - Supports the Change Request:
    - **Reason**: The Project Manager sees the feature as feasible but recommends added support for testing and a timeline buffer to mitigate the risks of staffing changes

Final outcome: With **majority support**, the recommendation is to proceed with the change request.

4.3. Change Request 3 - Government Reporting Compliance

Key: CR003

Description: Due to new government regulations, therapists are required to report patient services, including patient progress and timelines, monthly. This report must be submitted to a designated government server, ensuring compliance with the new mandate. The system needs to automatically compile, format, and submit these reports while maintaining data security and privacy standards.

Requirements:

- Generate a comprehensive report for each patient, including services received, progress, and timeline of attention.
- Submit the monthly report to the designated government server via a secure API endpoint.
- Implement required security measures to protect patient data during report compilation and submission.
- Ensure the data format aligns with government specifications.

Modules affected:

- **Patient Profiles:** Include fields to track additional information required for government reporting.
- **Progress Tracking:** Ensure all necessary data points for timelines and progress tracking are recorded for reporting.
- **Report Generation:** New module or extension to generate and format the monthly report.
- **API Integration:** Integration to securely submit reports to the government server.

Tasks:

Task	Estimation
Extend database to support additional patient report attributes	5 days
Develop reporting module to compile patient data into a government-compliant format	12 days
Create API integration for secure report submission	10 days
Implement logging for successful and failed submissions	4 days
Test the report generation and submission process	10 days

Feature costs:

Description	Days	Weeks	Cost per day	Total
Development	41	8.2	$1,200.00	$49,200
Testing	10	2	$800.00	$8,000.00
Management	8	1.6	$1,600.00	$12,800.00
Time Buffer	6	1.2	$1,200.00	$7,200.00
Total				**$77,200.00**

Internal cost:

Description	Weeks	Cost	Total
Development			
Mid Developer	4	$5753.42	$23,013.68
Junior Developer	4	$2,761.64	$11,046.56
Testing			
Tester	2	$3452.05	$6,904.10
Management			
Project Manager	1.6	$6443.83	$10,310.13
Total internal Cost			**$51,274.47**

Potential revenue:
Estimated potential revenue generated by this feature is **$77,200.00**. Yielding a 34% profit margin based on internal costs.

SWOT Matrix:

- Strengths
- Weaknesses
    - Adds workload amidst two other upcoming projects.
    - Team does not have the knowledge to implement government regulations.
- Opportunities
    - Builds trust and compliance standing with healthcare clients and government entities.

- Threats
    - Possible delays due to mid-developer reallocation and staffing changes.
    - In case that the change request is not implemented, the system won't be released. Because it does not comply with the government regulations.

CCB Criteria and Voting:

- CEO - Supports the Change Request
    - **Reason**: This feature is critical for regulatory compliance, enhancing the platform's value for healthcare clients. The profit margin is adequate, and the feature aligns with the company's goals to expand in the healthcare sector.
- Financial Department - Supports the Change Request
    - **Reason**: The profit margin is sustainable, with $77,200 billed to the client and internal costs estimated at $51,274.47, making it a financially sound addition.
- Human Resources - Supports the Change Request
    - **Reason:** This is a regulatory requirement, so HR supports the change. However, they advise monitoring workloads due to the departure of a tester and increased demands from new projects.
- Project Manager - Supports the Change Request
    - **Reason**: The Project Manager supports the feature, given that timelines are buffered and that additional resources for testing are secured if the team cannot cover the workload internally.

Final outcome: With **majority support**, the recommendation is to proceed with the change request.

4.4. Change Request 4 - Client Satisfaction Survey Module

Key: CR-004

Description: The client requests a new module dedicated to collecting client satisfaction data. This module will deploy random surveys to patients, gathering feedback about their rehabilitation experience. Based on the survey responses, the system should generate statistical insights. These insights should be available monthly or upon request to help professionals and management gauge satisfaction levels and identify areas for improvement.

Requirements:

- Randomly distribute satisfaction surveys to patients following sessions or at defined intervals.
- Include a variety of questions assessing patient satisfaction with the treatment, therapist interaction, and overall experience.
- Securely collect and store survey responses within the patient's profile.
- Ensure all responses are anonymized and follow data privacy standards.
- Generate monthly reports summarizing survey responses to highlight satisfaction trends.
- Provide real-time statistics accessible upon request, showing breakdowns by factors such as patient demographics, treatment types, and therapist interaction ratings.
- Add a section to the dashboard that displays key satisfaction metrics, trends, and average ratings.
- Allow professionals to view detailed survey insights for specific patient groups or timeframes.

Modules affected:

- **Patient Profiles:** Extend patient profiles to log survey participation and anonymized responses.
- **Survey Management Module:** New module to manage survey content, distribution schedules, and response collection.
- **Statistics & Analytics:** Expand statistics capabilities to generate satisfaction insights based on survey data.
- **Dashboard:** Integrate satisfaction statistics and insights to allow quick access for healthcare professionals.

Tasks:

Task	Estimation
Develop survey management module for creating and scheduling surveys	12 Days
Extend database for storing	6 days

anonymized survey responses	
Implement survey distribution logic for random patient sampling	8 days
Develop statistical analysis module for monthly and on-demand reporting	10 days
Add satisfaction metrics section to dashboard	5 days
Test survey distribution, response collection, and reporting functions	7 days

Feature costs:

Description	Days	Weeks	Cost per day	Total
Development	48	9.6	$1,200.00	$57,600.00
Testing	15	3	$800.00	$12,000.00
Management	12	2.4	$1,600.00	$19,200.00
Time Buffer	5	1	$1,200.00	$6,000.00
Total				**$94,800.00**

Internal cost:

Description	Weeks	Cost	Total
Development			
Mid Developer	5	$5753.42	$28,767.10
Junior Developer	5	$2,761.64	$13,808.20
Testing			
Tester	3	$3452.05	$10,356.15
Management			
Project Manager	2.4	$6443.83	$15,465.20
Total internal Cost			**$68,396.65**

Potential revenue:
Estimated potential revenue generated by this feature is **$94,800.00.** Yielding a 27.9% profit margin based on internal costs.

SWOT Matrix:

- Strengths
- Weaknesses
    - Survey distribution and analytics may require substantial testing to ensure quality results without disrupting user experience.
- Opportunities
    - Helps the client address patient needs proactively, positioning the application as a responsive healthcare tool.
    - Adds value to the application by providing client insights, enhancing client experience through data-driven adjustments.
- Threats
    - Complexity in survey deployment and analytics may affect timeline, especially with current workload and staffing constraints.

CCB Criteria and Voting:

- CEO - Does not support the Change Request
    - **Reason**: The future's alignment with strategic goals is noted, but additional profitability or cost-saving measures would be required to reconsider.
- Financial Department - Does not support the Change Request
    - **Reason**: While the feature is potentially valuable to the client, the profit margin of 27.9% is insufficient based on internal policy.
- Human Resources - Does not support the Change Request
    - **Reason:** The project could further strain resources, especially with new projects in the pipeline
- Project Manager - Supports the Change Request
    - **Reason**: Manager believes the feature is achievable and valuable with additional resource support or cost adjustments.

Final outcome: With majority **non-support**, the **recommendation is not to proceed** with Change Request 4 unless profitability can be adjusted to meet the 30% threshold.

5. Status Accounting

Status Accounting is the process of recording and reporting all changes made to the software throughout its lifecycle. This ensures traceability, transparency, and alignment with project goals by providing an overview of the current status of configuration items, change requests, and deliverables. All changes made are described in the Status Accounting Report using the next elements.

- **Date of Approval:** The date on which a change request was officially approved by the Change Control Board (CCB).
- **Date Implemented:** The date when the change request was fully implemented and integrated into the system.
- **CR Key**: A unique identifier assigned to each Change Request for easy tracking and reference (e.g., CR001).
- **CR Description:** A brief summary of the change request, detailing the purpose and scope of the proposed change.
- **CI Affected:** Specifies the Configuration Items (modules, databases, APIs, etc.) impacted by the change request.
- **Estimated Budget:** The predicted cost of implementing the change, including development, testing, and management efforts.
- **Estimated Time:** The anticipated duration required to complete the implementation of the change.
- **Estimated People:** The number of team members expected to work on the change request, including their roles.
- **Performed Budget:** The actual cost incurred during the implementation of the change request.
- **Performed Time:** The actual time taken to implement the change, compared to the estimated time.
- **Involved People:** The specific team members who contributed to the change request, including their roles and responsibilities.
- **Performance of Human Resources:** Evaluation of team performance during the implementation, including efficiency and productivity metrics.
- **Issues:** Any problems or challenges encountered during the implementation of the change request.
- **Outcome:** The final result of the change request, such as “Approved,” “Rejected,” or “Deferred,” along with its impact on the project.

5.1. Change request report 1 - Multi-location Clinic

- **Date of Approval:** November 10, 2024
- **Date Implemented:** Expected by January 5, 2025
- **CR Key:** CR001
- **CR Description:** Adds functionality to handle clinic-specific operations.
- **CI Affected:** Authentication, Patient Profile Management, Dashboard, Progress Tracking, API Integration
- **Estimated Budget:** $72,800
- **Estimated Time:** 6.8 weeks (including buffer)
- **Estimated People:** 4 (Mid Developer, Junior Developer, Tester, Project Manager)
- **Performed Budget:** $70,500
- **Performed Time:** 6.5 weeks
- **Involved People:** Mid Developer, Junior Developer, Tester, Project Manager
- **Performance of Human Resources:** Excellent; team completed tasks slightly ahead of schedule, with cost savings achieved through efficient resource allocation.
- **Issues:** Minor delays in API integration testing due to unanticipated data mapping challenges.
- **Outcome:** Approved; implemented as scheduled.

5.2. Change request report 2 - Other paralysis support

- **Date of Approval:** November 15, 2024
- **Date Implemented:** Expected by February 20, 2025
- **CR Key:** CR002
- **CR Description:** Adds support for new functionalities to include arm and leg paralysis, including custom routines, progress tracking, and dashboard updates.
- **CI Affected**: Patient Profile, Custom Routine Creation, Progress Tracking, Dashboard, API Integration
- **Estimated Budget:** $114,400.00
- **Estimated Time:** 12 weeks (including a 2-week buffer)
- **Estimated People:** 4 (Mid Developer, Junior Developer, Tester, Project Manager)
- **Performed Budget:** $110,500.00
- **Performed Time:** 11.5 weeks
- **Involved People:** Junior Developer, Tester, Project Manager (Mid Developer transitioned out during the project)
- **Performance of Human Resources:** Good; junior developer successfully stepped up to cover tasks, but testing resources were strained due to staffing limitations.

- **Issues:**
    - Tester resignation midway through the project delayed some testing tasks.
    - Mid Developer’s reassignment required additional support from junior resources, extending development timelines slightly.
- **Outcome:** Approved; implemented successfully but required resource adjustments and timeline optimization.

5.3. Change request report 3 - Government Reporting Compliance

- **Date of Approval:** November 15, 2024
- **Date Implemented:** Expected by February 20, 2025
- **CR Key:** CR002
- **CR Description:** Adds support for new functionalities to include arm and leg paralysis, including custom routines, progress tracking, and dashboard updates.
- **CI Affected**: Patient Profile, Custom Routine Creation, Progress Tracking, Dashboard, API Integration
- **Estimated Budget:** $114,400.00
- **Estimated Time:** 12 weeks (including a 2-week buffer)
- **Estimated People:** 4 (Mid Developer, Junior Developer, Tester, Project Manager)
- **Performed Budget:** $110,500.00
- **Performed Time:** 11.5 weeks
- **Involved People:** Junior Developer, Tester, Project Manager (Mid Developer transitioned out during the project)
- **Performance of Human Resources:** Good; junior developer successfully stepped up to cover tasks, but testing resources were strained due to staffing limitations.
- **Issues:**
    - Tester resignation midway through the project delayed some testing tasks.
    - Mid Developer’s reassignment required additional support from junior resources, extending development timelines slightly.
- **Outcome:** Approved; implemented successfully but required resource adjustments and timeline optimization.

6. Auditing

The Auditing section ensures that the project adheres to defined standards, requirements, and practices throughout its lifecycle. Auditing is critical for verifying the quality, security, and compliance of both processes and deliverables. Below are the key aspects of auditing for this project.

1. **Ensure that baselines and standards match the actual components of the product.**
    a. The latest version of the baseline aligns with the approved CRs for this project. CR-03 was fully implemented, including updates to the design, code, and tests. The other CRs (CR-01, CR-02, CR-04) were either rejected or deferred and are not reflected in the baseline.
2. **Verify that the product is built and documented as per the standards.**
    a. All process documentation, including the baseline, change request control document, and status accounting report, is complete and up-to-date. This ensures high-quality deliverables with traceable changes and proper records.
3. **Check if the CR and test cases were properly addressed.**
    a. CR-03 was implemented in accordance with the SCM process. All related test cases were executed and documented, ensuring alignment with the project's requirements.
4. **Check that the new test results added by the CRs give an acceptable result.**
    a. All test cases, including those designed for CR-03, passed successfully. No errors or issues arose from the implementation of this change.
5. **If tests fail, do…**
    a. Review the status accounting report.
    b. Refer to the configuration control document.
    c. Examine the configuration identification section in the configuration control document.

    These steps were unnecessary since all tests passed as expected.
6. **Verify that the correct and authorized versions of any CI exist and are correctly identified.**
    a. For this project, the Change Request Control document (v2.2) includes properly identified Configuration Items (CIs) for CR-03. All related updates have been tracked and validated.
7. **Verify that every CR on the Change Request Control document has a resolved status.**
    a. The Change Request Control document shows CR-01, CR-02, and CR-04 as rejected by the committee board, with only CR-03 approved and fully implemented.
8. **Identify the main biases or mistakes produced in each task.**
    a. Configuration Identification: The CI descriptions could benefit from more specificity to improve clarity for future updates.

   b. Configuration Control: The process is well-documented, but a slight reorganization of information would enhance readability.
   c. Status Accounting: The report is thorough but would benefit from an introductory section summarizing the SCM process.
9. **Check data quality for accuracy and completeness.**
   a. All documentation has been reviewed for clarity and completeness. Tasks and deliverables within each phase of the SCM process are well-documented, ensuring transparency and traceability.
10. **Initiate an incident for discovered unauthorized changes.**
   a. No unauthorized changes or incidents were identified during this auditing process.

7. Bibliography

[1] “What Is Configuration Management and Why Is It Important?,” IEEE Computer Society. https://www.computer.org/resources/software-configuration-management

[2]I. Buchanan, “Configuration management: definition and benefits,” Atlassian, 2024. https://www.atlassian.com/microservices/microservices-architecture/configuration-management

[3] Keyes, J. (2004). Software Configuration Management (1st ed.). Auerbach Publications. https://doi.org/10.1201/9780203496114