



## **Department of Software Engineering**

### **Software Configuration Management**

#### ***Audit Report for simple event registration system***

1. Kumneger Bekele – ETS 0955/14
2. Izzat Engida – ETS 0833/14
3. Jonathan Leoul – ETS 0834/14
4. Kaleab Hailemeskel – ETS 0847/14
5. Kaleab Mureja – ETS 0858/14
6. Leul Alemayehu – ETS 0961/14

**Submitted to:** Yimer (instructor)

Dec 2025, Addis Ababa, Ethiopia

## **Table of Contents**

<b>1.Introduction.....</b>	<b>2</b>
<b>2. Audit Scope and Methodology.....</b>	<b>2</b>
<b>3. Physical Configuration Audit.....</b>	<b>2</b>
<b>4. Functional Configuration Audit.....</b>	<b>3</b>
<b>5. Reflection.....</b>	<b>4</b>

## **1.Introduction**

The purpose of this report is to formally document the results of both the Physical Configuration Audit and the Functional Configuration Audit conducted on the project. These audits were performed in accordance with the approved Software Configuration Management Plan and standard configuration management practices.

The primary objective of the audit is to ensure that all configuration items are correctly identified, properly named, consistently versioned, and fully traceable, and that all approved change requests have been successfully implemented in the final system. This audit also serves as a final verification step to confirm that the system is ready for submission and release.

## **2. Audit Scope and Methodology**

The scope of this audit covered the entire Simple Event Registration System, including all source code files, documentation files, configuration management records, and change request documents. The GitHub repository served as the primary audit environment. The audit included verification of configuration items listed in the CI Register, confirmation of version numbers, inspection of naming standards, and validation of baseline tags and releases.

The Functional Configuration Audit focused on verifying whether the system behavior matches the approved functional requirements and whether all approved change requests were fully implemented. The audit process consisted of repository inspection, document cross-checking, manual execution of system features, and traceability verification between change requests, and the implemented code.

## **3. Physical Configuration Audit**

The Physical Configuration Audit was conducted to ensure that all documented configuration items physically exist in the repository and that they conform to the approved configuration standards defined in the SCMP. During the audit, the repository structure was examined and found to be consistent with the documented structure. The required folders for source files, documentation, and data files were all present and properly organized.

All configuration items listed in the CI Register were verified and confirmed to exist in their specified locations. This includes all HTML, CSS, JavaScript, JSON, and documentation files. The naming conventions were carefully reviewed, and it was confirmed that all configuration items followed the approved naming format such as DOC-\* for documents and CR-00X for change request forms. This confirms that naming discipline was consistently applied throughout the project lifecycle.

Version control was also carefully examined during the audit. Files that remained unchanged since the initial baseline were correctly maintained at version v1.0, while files that were modified due to approved change requests were properly updated to version v1.1. No version inconsistencies were detected, and no undocumented modifications were found.

The availability and correctness of essential SCM documents were also verified. The Software Configuration Management Plan, Configuration Item Register, Change Log, and all three approved Change Request forms were found in the documentation directory. All documents were properly versioned and consistent with the recorded project history.

Two baselines, BL1 and BL2, were found to be correctly created and tagged in the Git repository. These baselines correctly represent the original stable version and the updated version after change implementation. In addition, GitHub releases corresponding to version v1.0 and v1.1 were published successfully, and release notes accurately described the system state at each release.

Based on these observations, it was concluded that the Physical Configuration Audit was successfully completed with no discrepancies found. All configuration items are complete, properly controlled, and fully traceable to their documentation.

#### **4. Functional Configuration Audit**

The Functional Configuration Audit was conducted to verify that the implemented system satisfies all approved functional requirements and that all approved Change Requests were fully and correctly implemented. The FCA focused on validating the operational behavior of the Simple Event Registration System through direct system execution and testing.

The login functionality of the system was first tested using valid credentials. The system successfully authenticated the user and restricted access to the dashboard, confirming that the

security and access control requirement was correctly implemented. The event dashboard was then tested and confirmed to correctly display the list of available events loaded dynamically from the events.json data file. This verifies that the system correctly integrates external data into the user interface.

The event registration process was then tested. Users were able to register for events successfully, and the system responded correctly to registration actions. As part of Change Request CR-001, an animated success message was required after successful registration. This feature was verified visually and was found to function as expected.

Change Request CR-002 required the implementation of a “My Registrations” feature that allows users to view the list of events they have registered for. This functionality was verified by registering for multiple events and confirming that the registered events appeared correctly in the user’s personal registration section using localStorage as the storage mechanism.

Change Request CR-003 focused on implementing a seat availability counter. This feature was tested by repeatedly registering users for a selected event and observing the reduction in available seats. Once the available seats reached zero, the system correctly disabled the registration button and displayed the “Event Full” status.

All three Change Requests were also verified at the configuration management level. Each change request was implemented using a separate Git branch, reviewed through a pull request, and properly merged into the main branch. The Change Log accurately records all three changes, and traceability was established between the change request documents, the modified files, and the baseline update.

Based on functional testing and configuration verification, it was confirmed that all required system features are working correctly and that all approved changes have been fully implemented.

## **5. Reflection**

The Configuration Audit of the Simple Event Registration System was completed with no discrepancies. The Physical Configuration Audit confirmed that all configuration items are correctly named, versioned, and fully traceable to the project documentation, with baselines and releases properly maintained. The Functional Configuration Audit verified that all

required features and approved change requests were correctly implemented and met the assignment requirements. Overall, the project followed the SCMP effectively, with no unauthorized changes and strong adherence to proper configuration management practices.