**Automated Equipment Checkout System**

**Software Design Description**

Victor Camacho

Version Number: 1

December 1, 2024

Revisions

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Revisions | Description | Author |
| 11/1/2024 | 1.0 | Initial draft | Victor Camacho |

Contents

[1. Introduction 4](#_Toc183989357)

[1.1 Purpose 4](#_Toc183989358)

[1.2 Scope 4](#_Toc183989359)

[1.3 Definitions, Acronyms, Abbreviations 4](#_Toc183989360)

[1.4 References 4](#_Toc183989361)

[1.5 Overview 4](#_Toc183989362)

[2. System Overview 5](#_Toc183989363)

[3. System Architecture 5](#_Toc183989364)

[3.1 Architectural Design Summary 5](#_Toc183989365)

[3.2 Rationale For Selecting Layered Architecture 5](#_Toc183989366)

[3.3 Diagrams 6](#_Toc183989367)

[4. Data Dictionary 7](#_Toc183989368)

[5. Component Design 8](#_Toc183989369)

[5.1 Static View 8](#_Toc183989370)

[5.2 Dynamic View 9](#_Toc183989371)

[5.3 Human Interface Design 10](#_Toc183989372)

[6. Requirements Matrix 11](#_Toc183989373)

# Introduction

## Purpose

This document defines the Software Design Description (SDD) for the Automated Equipment Checkout System for GB Manufacturing. It provides a detailed design blueprint that guides the system's construction and implementation. It includes the architectural design, data structures, interfaces, and components necessary to achieve the system’s requirements.

## Scope

This project aims to develop a software-based automated equipment checkout system for GB Manufacturing. The **SDD** will provide a detailed design that enables the system to manage and track the check-in and check-out of tools and equipment, provide real-time tracking, and generate alerts for overdue returns. The design ensures the system is modular and scalable and adheres to quality attributes such as usability, reliability, performance, and supportability.

## Definitions, Acronyms, Abbreviations

* **SDLC** - Software Development Life Cycle: Process for software development phases.
* **SRS** - Software Requirements Specification: Document detailing system requirements.
* **GB Manufacturing** - The company for which the system is developed.
* **UML** - Unified Modeling Language: Diagrams representing system interactions and structure.
* **OOAD** - Object-Oriented Analysis and Design: Approach for designing software using objects.
* **VOPC Matrix** - View of Participating Classes Matrix: Table linking use cases to classes.
* **SDD - Software Design Description:** A document outlining the system's design details, including architecture, components, and interfaces.

## References

* Software Requirements Specification (SRS), Version 1.2

## Overview

This Software Design Description (SDD) provides the architectural and component design for the Automated Equipment Checkout System. It outlines the system's static and dynamic views, implemented design patterns, and the human interface considerations. This document guides developers and stakeholders in understanding the design and ensuring the system meets the requirements specified in the SRS.

# System Overview

The Automated Equipment Checkout System is a software solution designed for GB Manufacturing to improve the tracking and management of equipment within the maintenance department. It aims to reduce equipment loss and enhance accountability by providing a seamless and automated check-in and check-out process.

This system uses a **Layered Architecture** to separate concerns between the user interface, business logic, and data storage, ensuring better scalability, maintainability, and modularity. Key features include:

1. **User Login and Access Control:** Only authorized personnel can access the system.
2. **Real-Time Equipment Tracking:** Tracks the availability of tools and their current status, such as checked out, overdue, or available.
3. **Overdue Notifications:** Sends automated alerts for overdue returns to ensure accountability.
4. **Reporting Tools:** Provides administrative capabilities for generating usage and overdue reports.

The system supports GB Manufacturing’s operational goals by enabling real-time monitoring, streamlining equipment management processes, and reducing the financial impact of lost or unreturned equipment.

# System Architecture

## Architectural Design Summary

The Automated Equipment Checkout System utilizes a **Layered Architecture** with an optional **Tiered Approach** for added scalability and security. This design divides the system into distinct layers with the following responsibilities:

1. **Presentation Layer (User Interface Layer):**  
   Handles user interactions such as login, viewing equipment status, and generating reports.
2. **Business Logic Layer:**  
   Processes system logic, manages equipment checkouts/returns and enforces rules like overdue notifications.
3. **Data Layer:**  
   Manages data storage and retrieval for equipment, users, and transaction records.

## Rationale For Selecting Layered Architecture

* **Usability:** Provides a clean separation between interface design and business rules.
* **Reliability:** Minimizes system-wide failures by isolating responsibilities.
* **Performance:** Allows optimization at individual layers without impacting others.
* **Supportability:** Simplifies updates and maintenance due to modular design.

The **Tiered Architecture** (optional) adds further **scalability** by separating layers into independent physical servers and improves **security** by restricting direct access to data.

## Diagrams

* **UML Class Diagram**



* **UML Sequence Diagram**



# Data Dictionary

The **Data Dictionary** defines the key entities and elements in the Automated Equipment Checkout System, including their names, types, and descriptions. It ensures consistent understanding and usage of these elements throughout the system.

|  |  |  |
| --- | --- | --- |
| **Entity Name** | **Type** | **Description** |
| User | Class | Represents an employee accessing the system. |
| Admin | Class | Represents an administrator managing reports and overseeing system operations. |
| Checkout | Class | Manages the equipment checkout and return process. |
| Equipment | Class | Represents tools or equipment in the system, along with their status. |
| Report | Class | Handles the generation of usage summaries and overdue equipment reports. |
| userID | Integer | Unique identifier for each user in the system. |
| equipmentID | Integer | Unique identifier for each piece of equipment. |
| status | String | Indicates the current status of an equipment item (e.g., Available, Checked Out). |
| dueDate | Date | Specifies the due date for equipment returns. |

# Component Design

## Static View

The **UML Class Diagram** below represents the system's static structure, showing the relationships between key classes and their attributes and methods.



## Dynamic View

The **Dynamic View** illustrates the interactions between the system components during user authentication. The following sequence diagram demonstrates the process of validating a user's credentials and granting access to the system.

**Sequence Diagram for User Authentication**



**Description:**

The sequence diagram shows the steps in the authentication process:

1. The user enters login credentials through the user interface.
2. The system sends the credentials to the authentication service for validation.
3. The authentication service queries the database to verify the user details.
4. the authentication service returns a success or failure message based on the database response.
5. The user is notified of the login result through the interface.

## Human Interface Design

**User Interface Overview:**

The system provides a simple, intuitive user interface to streamline equipment checkouts and returns. Key components include:

1. **Login Screen**: Allows users to authenticate using unique credentials.
2. **Dashboard**: Displays an overview of available and checked-out equipment.
3. **Check-Out Screen**: Enables employees to search for equipment and record the check-out process.
4. **Check-In Screen**: Provides a form to return equipment and update its status.
5. **Notifications Panel**: Alerts users of overdue items and upcoming return deadlines.

**Interaction Design:**

1. **Input Mechanisms**:
   * Textboxes for user credentials on the login screen.
   * Dropdown menus to select equipment.
   * Submit buttons to complete check-out and check-in tasks.
2. **Output Mechanisms**:
   * Real-time equipment availability is displayed on the dashboard.
   * Confirmation messages after successful check-outs or check-ins.
   * Notifications for overdue items in a prominent alerts section.

# Requirements Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Requirement Description** | **Mapped Components** | **Mapped Interfaces/Views** |
| REQ-001 | The system shall authenticate users with unique credentials to ensure secure access when they log in. | User Class | User Login Sequence Diagram (Dynamic View) |
| REQ-002 | The system shall allow employees to check out equipment, updating the status to “in use” to reflect current availability. | Checkout Class, Equipment Class | Check-Out Equipment Sequence Diagram (Dynamic View) |
| REQ-003 | The system shall allow employees to check equipment, updating the status to “available” for future use. | Checkout Class, Equipment Class | Update Equipment Status Dynamic View |
| REQ-004 | The system shall provide real-time status updates for each piece of equipment, indicating whether it is “available,” “in use,” or “overdue.” | Equipment Class, Report Class | Static Diagram (Class Relationships) |
| REQ-005 | To ensure accountability, the system shall automatically notify employees and managers when equipment is overdue for return. | Checkout Class, Report Class | Notify Overdue Sequence Diagram (Dynamic View) |
| REQ-006 | The system shall generate usage reports for management, highlighting trends and patterns in equipment usage. | Admin Class, Report Class | Report Generation Sequence Diagram (Dynamic View) |
| REQ-007 | The system shall display a list of all available equipment, allowing employees to view and select items for checkout. | Equipment Class | Static Diagram |
| REQ-008 | The system shall log each check-in and check-out transaction to maintain a complete record for tracking and accountability. | Checkout Class, Equipment Class | Static Diagram, Check-Out Equipment Sequence Diagram |