Uzair Cotwall, Mufhatutshedzwa Todani, Halalisani Asanda

Mdlalose, Ndumiso Bandile Zwane

Student numbers: ST10092380, ST10119248, ST10092007,

ST10092443

PATHWAY: XBCAD7319

Lecturer: Handsome Mpofu

29 November 2023

Work Integrated Learning 3: POE

Introduction to the project

Project Overview:
The project entails the development of a software solution aimed at enhancing the management of Personal Protective Equipment (PPE) within Parktown Manufacturers. Presently, the PPE distribution process relies on manual Excel-based procedures, resulting in inefficiencies, errors, and a lack of historical data accessibility. The software's primary goal is to streamline this process by introducing a digital platform for PPE distribution, ensuring precise record-keeping, and simplifying historical data retrieval.

Background:
Within Parktown Manufacturers, there are around 80 employees distributed across various departments, and approximately 20 different types of PPE items are in use. Each week, the Distribution Officer is tasked with manually recording the following details for each PPE item issued to employees:

- PPE ITEM TAKEN
- HAS THE ITEM BEEN TAKEN BEFORE
- DATE OF LAST ISSUE
- NAME
- DEARTMENT
- SIGNATURE

Additionally, different departments have varying allocation rules for specific PPE items, introducing complexity into the process. The existing manual system presents challenges such as inefficiency, potential data entry errors, and the need for manual historical data calculations. The software solution is intended to overcome these challenges by automating PPE tracking and reporting.

Work Agreement:

Project Overview:

The Parktown Manufacturers PPE Management Software Project intends to create a digital solution to improve the organization's management of the distribution of Personal Protective Equipment (PPE). The program will take the place of the current manual Excel-based method, improving accessibility to historical data and increasing efficiency. The scope, goals, roles, and duties

necessary to carry out this project successfully are described in this work agreement.
The project's scope includes the following:

1. creating a digital distribution channel for PPE.
2. automating the documentation of PPE distribution information, such as:
• PPE item removed.
• Has the item already been taken?
• Last issue date.
• Department of the employee Name of the employee.
• Employee's signature.
3. Putting allocation guidelines for PPE goods into effect for various departments.
4. Ensuring accurate record-keeping and the retrieval of past data.
5. Addressing the 20 different types of PPE equipment and the needs of about 80 people.

Project goals:
The following are the project's main goals:

1. distributing PPE more efficiently.
2. reducing data entry mistakes.
3. automating calculations using past data.
4. increasing the management of PPE's overall effectiveness.
5. improving public access to past PPE data.

Responsibilities and Roles:

1. **Project Sponsor (Parktown Manufacturers):**
• Provide the project with resources and funding it needs.
• Approve project milestones and deliverables.

2. **Project Director:**

• Lead the project team.
• Create and maintain the project plan.
• Make sure the project's goals are attained on schedule and within budget.
• Inform the project sponsor and other stakeholders about the status and any problems.

3. **Team for Development:**

• Design, develop, and test the PPE Management Software.
• Verify if the software satisfies the specifications.
• Work together with end users to test and gather feedback.

4. **Officer of Distribution (End-User):**

• Work along with the development team to specify the distribution policies for PPE.
• During user acceptance testing, provide feedback.
• Assist with the software training for employees.

Definition of Ready:

Before a job is deemed "ready" for execution inside the project, it must first satisfy the following requirements:

1. **Clear Requirements:**
• The task's specific needs are outlined and authorized.
2. **Identified Resources and Dependencies:**
• There is access to all required employees, equipment, and data.
• All external dependencies are noted and taken care of.
3. **Acceptance Standards Defining:**
• When the task is finished, there are set clear acceptance requirements.
4. **Prioritization and Estimation:**
• Estimates of the work's duration and effort are given.
• The project schedule is determined in alignment with the task priority.
5. **Sharing of knowledge and documentation:**
• The team has access to all necessary information and documents to complete the task.

We intend to expedite project execution and raise the possibility of successful delivery by adhering to this work agreement and making sure that tasks comply with the Definition of Ready.

Definition of Done:

Definition of Done (DoD): The Definition of Done (DoD) serves as the comprehensive set of criteria that each user story and feature related to PPE management must satisfy to be deemed complete and ready for deployment. The DoD ensures that the software solution attains the highest quality standards and effectively caters to the needs of the Distribution Officer and the organization.

The DoD encompasses, without limitation, the following criteria:

1. **Functional Requirements:** All specified functionalities, as stipulated in user stories and epics, must be implemented, and rigorously tested to confirm alignment with the intended objectives.
2. **Usability and User Experience:** The software should furnish an intuitive and user-friendly interface, simplifying the PPE management process for the Distribution Officer.
3. **Security and Data Privacy:** All user data, including employee profiles and PPE issuance records, must be securely stored and shielded from unauthorized access to ensure data confidentiality and compliance with privacy regulations.
4. **Performance and Scalability:** The software must remain responsive and maintain excellent performance even with an increased number of employees and PPE items to guarantee a seamless user experience.
5. **Reliability and Availability:** The system must possess high availability with minimal downtime during regular working hours, safeguarding uninterrupted PPE distribution operations.
6. **Compliance and Reporting:** The software needs to facilitate the generation of compliance reports for verifying adherence to PPE allocation rules, catering to auditing and accountability requirements.
7. **Notification Responsiveness:** Notifications, including alerts for nearing allocation limits, must be dispatched promptly to the Distribution Officer to uphold compliance and employee safety.
8. **Scalable Database:** The database employed for storing PPE issuance records should exhibit scalability to accommodate long-term data storage and retrieval.
9. **Accessibility:** The software must adhere to pertinent accessibility standards, ensuring effective usage by all, including individuals with disabilities.
10. **Documentation and Training:** The provision of comprehensive user documentation and training materials is imperative to assist the Distribution Officer in the effective utilization of the software and to diminish the learning curve.

Adhering to these criteria in the DoD will substantiate that the PPE management software attains superior quality standards and aligns seamlessly with the organization's strategic objectives.

RoadMap (High-Level):

	Project Start date	28-Aug-23				
	Project Name	PPE ProTrack				Week starting
#	Activity	Start	End	Days	Status	%Done
1	**Project Initiation Phase**					
1.1	Project Kick-off	28-Aug-23	28-Aug-23	1	Complete	100%
1.2	Work Agreement, DoR, DoD, and High-Level Plan	29-Aug-23	1-Sep-23	4	Complete	100%
2	**Requirements Gathering and Analysis**					
2.1	User Roles	4-Sep-23	5-Sep-23	2	Complete	100%
2.2	User Stories	5-Sep-23	7-Sep-23	3	Complete	100%
2.3	User Experience Journey Map	7-Sep-23	10-Sep-23	2	Complete	100%
3	**Planning and Design Phase**					
3.1	Architecture and Design	11-Sep-23	20-Sep-23	8	In Progress	10%
3.2	Technology Stack Selection	21-Sep-23	25-Sep-23	3	Not Started	0%
3.3	Development Plan	26-Sep-23	30-Sep-23	4	Not Started	0%
4	**Development Phase**					
4.1	Iterative Development	1-Oct-23	17-Nov-23	35	Not Started	0%
5	**Testing and Quality Assurance Phase**					
5.1	User Acceptance Testing	20-Nov-23	24-Nov-23	5	Not Started	0%
5.2	Performance Testing	25-Nov-23	27-Nov-23	1	Not Started	0%
6	**Deployment and Ongoing Improvements Phase**					
6.1	Deployment to Production	29-Nov-23	1-Dec-23	3	Not Started	0%
6.2	Post-launch Support	1-Dec-23	30-Dec-23	21	Not Started	0%
				92		

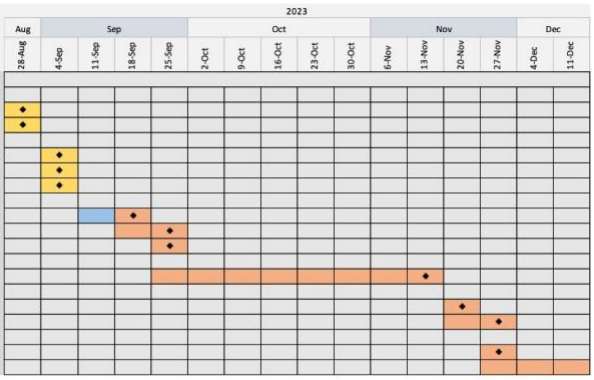

Figure 1: Roadmap (High-Level plan)

Requirements

(Spreadsheets provided by the distribution officer are provided in the folder with the following names:

- 1. Daily Report
- 2. Annalysis Weekly usage report MAIN

)

User Roles

Distribution Officer

- Role Description: The Distribution Officer is the primary user of the PPE management software. Their responsibilities include distributing PPE items to employees, recording issuance details, and maintaining accurate records.

User Stories (Epic)

User Stories:

1. As a Distribution Officer, I want to log in securely to the system with my credentials.
2. As a Distribution Officer, I want to create and manage employee profiles, which includes their department information and rules for allocating Personal Protective Equipment (PPE).
3. As a Distribution Officer, I want to add new PPE items to the system providing descriptions and specifying allocation rules.
4. As a Distribution Officer, I want to issue PPE items to employees, and record issuance details such as date, name, and usage duration.
5. As a Distribution Officer, I want to view the history of PPE issuances for each employee.
6. As a Distribution Officer, I want the system to automatically calculate how many PPE items are still available for each employee.
7. As a Distribution Officer, I want to generate reports on PPE usage, including the total number of items issued and the usage duration.
8. As a Distribution Officer, I want to receive notifications when an employee approaches the limit of their allocated PPE items.

User Experience Journey Map

User Journey Map: Distribution Officer's PPE Management

Parktown Manufacturer Experience Map

This experience map visualizes the ideal user experience, specifically for the distribution officer . This map will serve as our guide as we design the product, particularly the first prototype.

Stages	Getting Started	Setting up Employees	Configuring PPE Items	Distributing PPE	Retrieving Records	Analyzing Data	Staying Informed	Exporting Data
Key Actions	• Distribution Officer logs in for the first time. • Officer sets up their profile and department information.	• Adding new employees to the system. •Reviewing and updating employee profiles	• Adding new PPE items to the system. • Maintaining an updated list of available PPE items.	• Selecting an employee. • Choosing the PPE item to issue (quantity, date, and usage duration). • System auto-calculation of remaining quantities.	• Viewing issuance history for a specific employee.	• Generating reports for PPE usage analysis.	• Receiving notifications when an employee nears allocation limits.	• Exporting PPE issuance data for long-term storage and auditing.
Doing	• Enter personal information and preferences	• Inputting employee details. • Ensuring accurate data.	• Defining item attributes (e.g., descriptions, allocation rules). •Regularly updating the PPE inventory.	• Ensuring accurate issuance records. • Monitoring compliance with allocation rules.	• Analyzing historical data. • Checking for discrepancies or trends.	• Reviewing reports for insights. • Making data-driven decisions.	• Promptly addressing allocation issues. • Monitoring for alerts.	• Ensuring data is securely archived. • Completing the export process efficiently.
Touchpoints	- Login page - user profile setup	- Employee profile management interface	- PPE item configuration interface	- PPE issuance interface - employee selection	- PPE history view	- Reporting and analytics tools.	- Notification system	- Data export functionality.
Emotions and Thoughts	• Curious and eager to get started. • Hoping for a user-friendly and efficient system. • Slightly concerned about the learning curve	• Need for accuracy in employee data. • Concerns about data entry errors.	• Ensuring the accuracy of PPE item details. • Streamlining the process to save time.	• Focus on accurate recording. • Ensuring compliance with allocation rules. • Appreciation for the system's auto-calculation feature.	• Need for quick access to historical data. • Checking for any discrepancies. • Appreciation for a clear history log.	• Desire for easy report generation. • Analyze data for decision-making. • Look for insights to improve allocation rules.	• Relief at having a reminder system. • Prompt action to maintain compliance. • Positive experience when notifications are helpful.	• Easy and reliable export process. • Ensuring data is securely stored for compliance purposes.
Emotions	Excited	Neutral	Neutral	Neutral	Happy	Neutral	Happy	Happy

Turtle

miro

Figure 2: User Experience Journey Map

This User Experience Journey Map outlines the steps and interactions that the Distribution Officer will have while using the PPE management software. It ensures a user-centric approach to software development and aims to provide a smooth and efficient experience for the officer in charge of distribution.

Non-Functional Requirements

The following non-functional requirements have been gathered through interactions with the client, the Distribution Officer, and are essential to the success of the PPE management software project:

1. Usability and User Experience:
- Requirement: It is essential for the software to have a user-friendly interface.
- Rationale: We need to prioritize ease of use so that even someone, without skills like the Distribution Officer can efficiently manage PPE.

2. Security and Data Privacy:
- Requirement: It is crucial that all user data, including employee profiles and records of PPE issuance is securely stored and protected from access.
- Rationale: Compliance, with data privacy regulations relies heavily on ensuring the confidentiality and integrity of PPE distribution data.

3. Performance and Scalability:
Requirement: The software should be fast and efficient even as the number of employees and PPE items increases.
Rationale: As the organization grows the software needs to handle data without any decrease, in performance.

4. Reliability and Availability:
Requirement: The system should have availability and minimal downtime during working hours.
Rationale: It is crucial to have uninterrupted access to the software for PPE distribution operations.

5. Compliance and Reporting:
Requirement: The software must support generating compliance reports to ensure adherence to PPE allocation rules.
Rationale: Compliance reports are essential for auditing purposes. Maintaining accountability.

6. Notification Responsiveness:
Requirement: Distribution Officer notifications, like alerts for approaching allocation limits should be delivered promptly.
Rationale: Timely notifications play a role, in upholding compliance standards. Ensuring employee safety.

7. Scalable Database:
Requirement: The database used for storing PPE issuance records should be able to handle long term data storage needs.

Rationale: Over time the database will accumulate an amount of data that needs to be easily retrievable and manageable.

8.
Accessibility:
Requirement: The software must be designed to cater to users with disabilities adhering to the accessibility standards.
Rationale: It is important to make sure that all users regardless of disabilities can effectively utilize the software.

9. Documentation and Training:
- Requirement: Comprehensive user documentation and training materials should be provided to assist the Distribution Officer in using the software effectively.
- Rationale: Proper documentation and training reduce the learning curve and promote user adoption.

Demonstrate understanding of analysis

Identify Bounded Context

User Management Bounded Context:

Responsible for managing user authentication and authorization. Includes functionalities related to user login, password management, and user roles and permissions.

Employee Profile Bounded Context:

Manages employee information, including department details and allocation rules. Functions related to creating, updating, and deleting employee profiles fall within this context.

PPE Item Management Bounded Context:

Focuses on managing PPE items, including their descriptions, allocation rules, and availability. Handles operations like adding new PPE items and defining allocation guidelines for different departments.

PPE Issuance Bounded Context:

Manages the process of issuing PPE items to employees. Includes functionalities for recording issuance details, such as date, employee name, department, and usage duration.

Reporting and Notification Bounded Context:

Deals with generating reports related to PPE usage, compliance, and allocation statistics. Manages notifications, including alerts for approaching allocation limits and compliance violations.

Compliance and Rules Bounded Context:

Enforces compliance with PPE allocation rules specified for different departments. Manages rules regarding the maximum number of PPE items allowed per employee in various departments.

Define Bounded Context

In software design, Bounded Contexts are a key concept in Domain-Driven Design (DDD). They define explicit boundaries within which a particular domain model and the associated concepts and terms are valid. Bounded Contexts help manage the complexity of large software systems by breaking them down into smaller, more manageable components, each with its own specific responsibility and understanding of the domain.

Here are the core aspects of Bounded Contexts:

1. Clear Boundaries:

Explicit Limits: Bounded Contexts have clearly defined interfaces and explicit boundaries, separating one context from another.
Scope: A Bounded Context encapsulates a specific part of the domain, containing entities, value objects, aggregates, and domain logic relevant only to that context.

2. Unique Language:

Ubiquitous Language: Each Bounded Context has its own language and terminology that might differ from other contexts. Terms and concepts used inside a Bounded Context have specific meanings unique to that context.

3. Isolation of Models:

Separate Models: Different Bounded Contexts might have different interpretations of the same entity or concept. For example, "Customer" in one context could mean something different from "Customer" in another context.

4. Autonomous Development:

Independent Development: Bounded Contexts can be developed, deployed, and versioned independently. Changes within one Bounded Context should not directly affect other contexts.

5. Context Mapping:

Context Mapping: When Bounded Contexts need to communicate, explicit mappings and translation layers define how data and interactions are shared between contexts. Context mapping ensures consistency and meaningful communication between contexts.

6. Microservices and Bounded Contexts:

Microservices Alignment: Bounded Contexts are often aligned with microservices in a microservices architecture. Each microservice represents a Bounded Context, allowing for modular development, deployment, and scaling of services.

7. Problem-Specific:

Problem-Specific Solutions: Bounded Contexts allow developers to tailor solutions specific to the problems within that context. The domain logic inside a context is crafted to meet the unique requirements of that part of the domain.

8. Consistency Within, Flexibility Across:

Consistency Within: Within a Bounded Context, the data and logic maintain internal consistency.
Flexibility Across: Bounded Contexts provide flexibility and adaptability across the entire system, allowing for independent evolution and changes.

UML Package Diagram

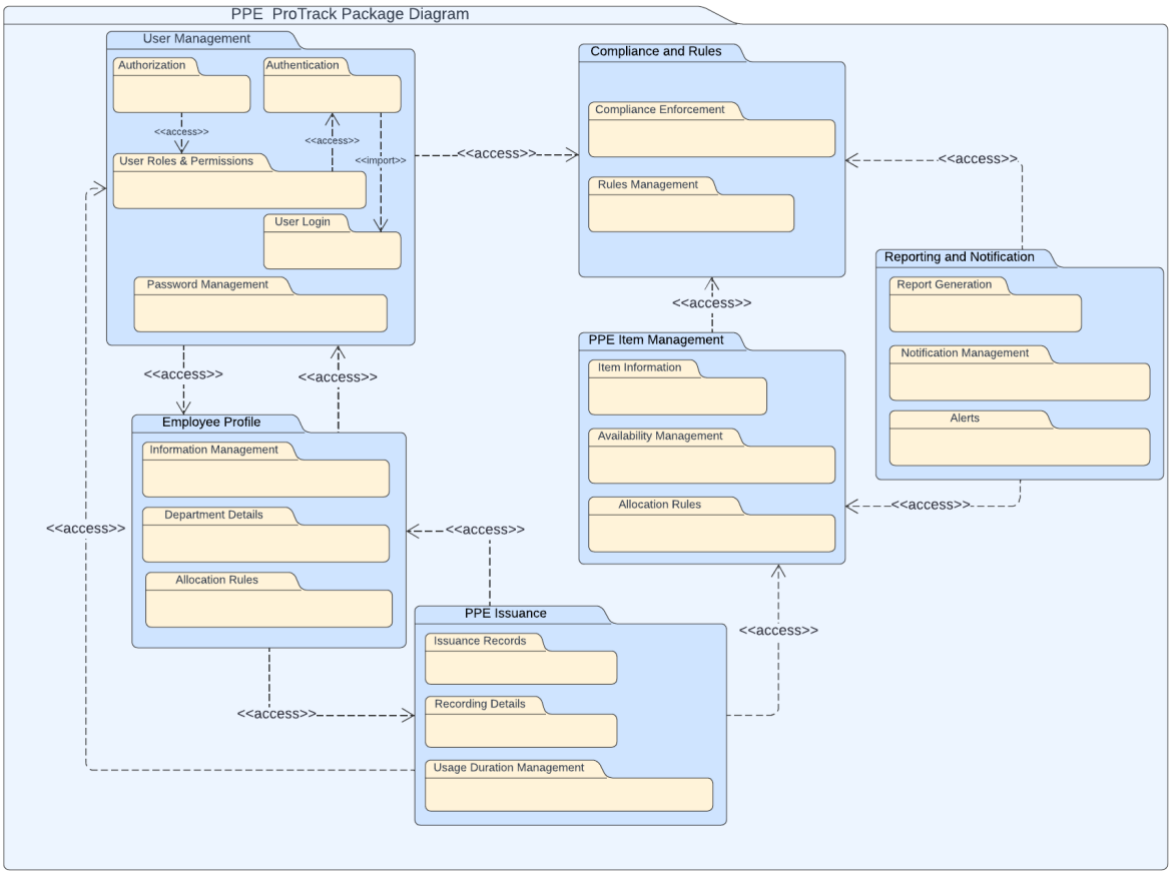

Figure 3 UML Package Diagram (Visual Paradigm, (n.d)b)

Demonstrate understanding of software implementation documentation

UML Object Diagram

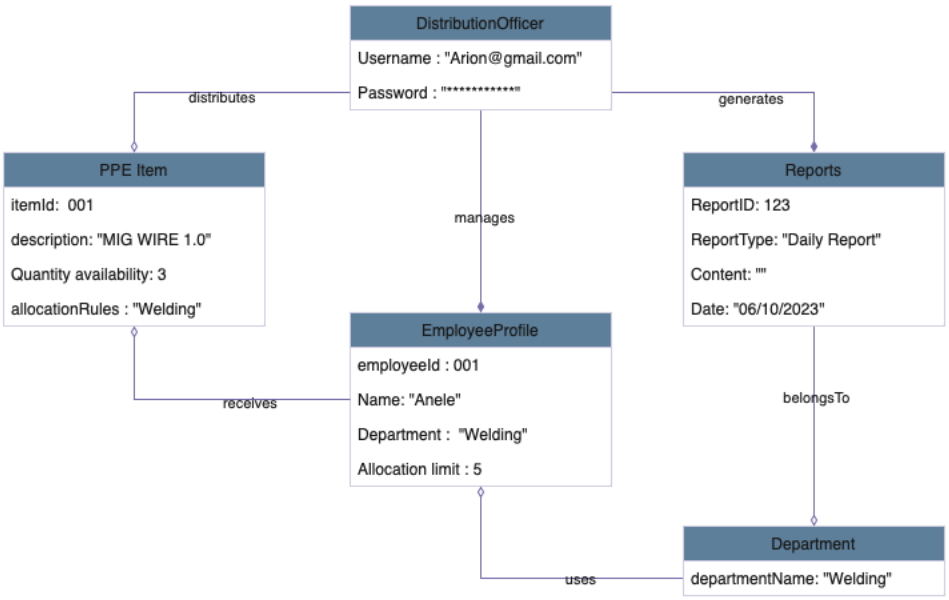

Figure 4 Object Diagram (Visual Paradigm, (n.d)a; Plantuml, (n.d))

UML sequence diagram

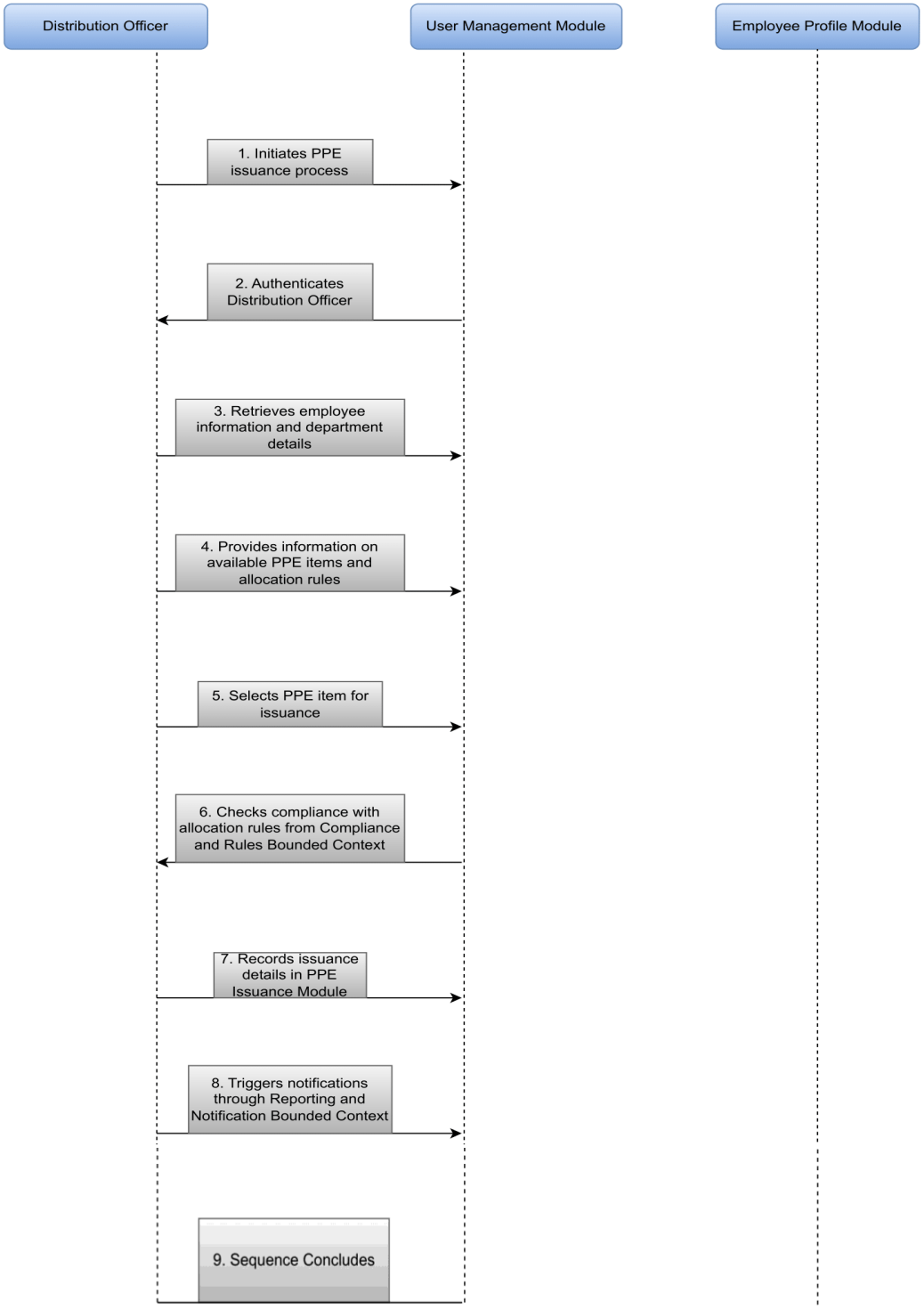

Figure 5 Sequence diagram (Creately, 2022a; Creately, 2022b)

UML State diagrams

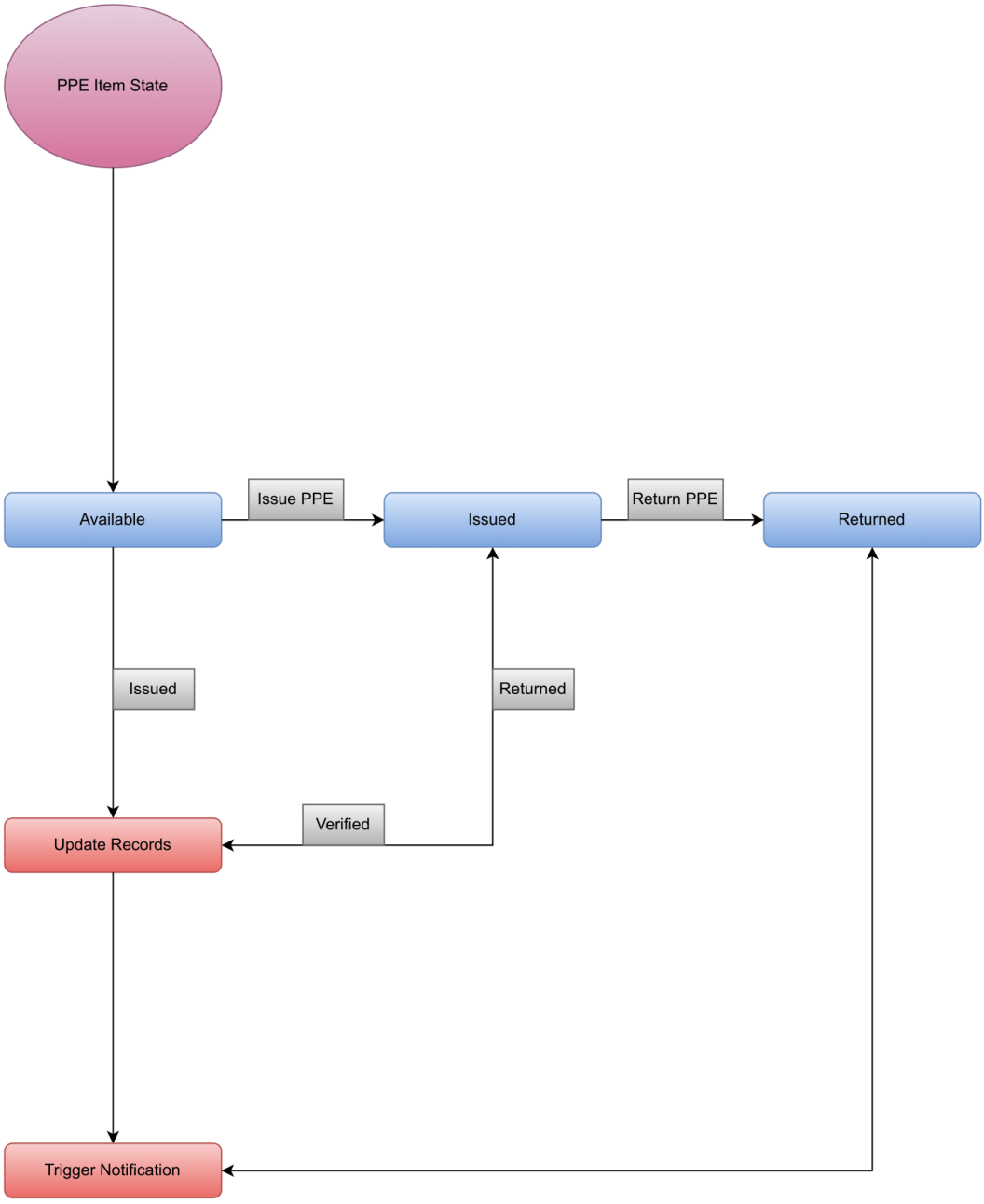

Figure 6 UML State diagram (Guthrie, 2022)

Demonstrate understanding of data schemas

Data Storage Description:

The success of our project relies heavily on how we manage our data storage. Storing data, within the system entails the preservation of employee profiles, PPE items, and issuance records. This process includes making use databases to organize this information. Data storage involves processes and technologies that securely capture, store, and organize the data essential for our PPE management system and employee records. (bmc, 2020)

Here is a detailed overview of our strategy for storing data:

Data Types:

Our project deals with various data types:

- Employee Information: This encompasses employee profiles that store information such, as employee ID, first name, last name, department, and email address.
- PPE Inventory: We maintain records of protective equipment (PPE) items, which include the names, descriptions, and quantities of each item.
- PPE Issuance Records: We maintain a record of equipment (PPE) items provided to employees, which includes the date of issuance, return dates and the quantities issued.
- Department Information: this would include all the departments, in the organization including their names and the individuals who manage them.

Data Source:

- Manual Data Entry: The Distribution Officer manually enters information into the system including details, about the issuance of Personal Protective Equipment (PPE) and employee information.

Data Storage Technologies:

To accommodate our data storage needs effectively, we have chosen the following technologies:

- Firebase: we will use Firebase, which is a cloud-based platform to handle our data storage needs. It provides us with a scalable solution that allows us to efficiently store and retrieve data while ensuring its consistency, reliability, and accessibility.

Key Data Entities:

The key data entities in our system and their significance include:

- Employee: Employee data plays a role, in our system as it assists in monitoring the distribution of PPE items implementing department allocation guidelines and facilitating efficient communication.
- PPE Items: This entity represents protective equipment (PPE) items, including item name, description, and quantities.
- PPE Issuance Records: This entity keeps a record of the distribution and retrieval of PPE supplies maintaining a log of who used which items and when. It helps ensure compliance facilitates reporting and enables auditing processes.

- Department: Departments play a role, in ensuring that personal protective equipment (PPE) is distributed appropriately to individuals according to their roles and specific departments.
-

ERD

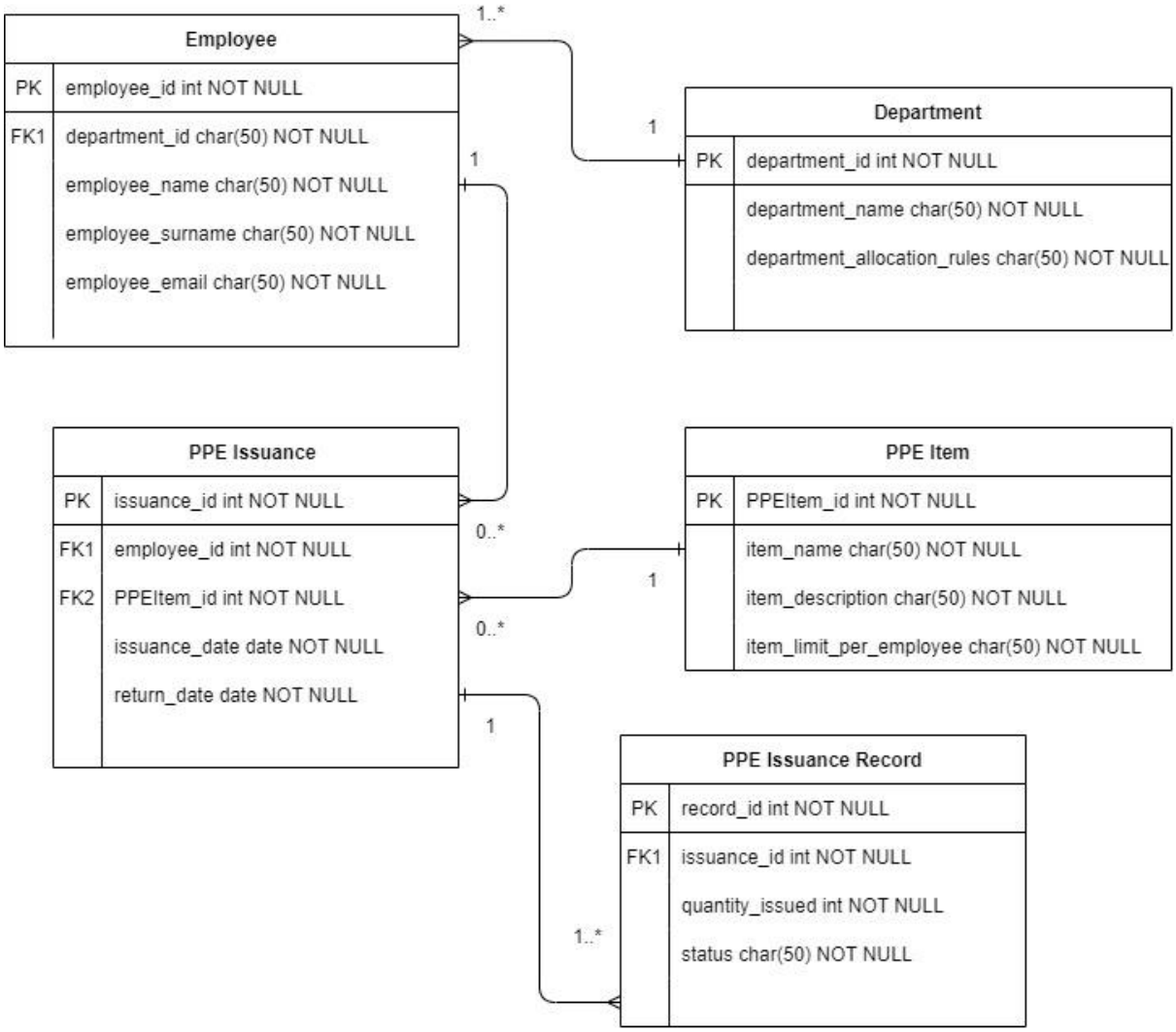

Figure 7 domain model Entity Relationship Diagram (Lucidchart, (n.d.))

Relationships:

- One Employee can belong to One Department (many-to-one).
- One Department can have Many Employees (one-to-many).
- One PPE Issuance is associated with One Employee and One PPE Item (many-to-one).
- One Employee can have Many PPE Issuances (one-to-many).
- One PPE Item can be associated with Many PPE Issuances (one-to-many). Multiple different items can be issued in one issuance event for one employee.
- One PPE Issuance can have Many PPE Issuance Records (one-to-many). In one issuance event one employee might get issued multiple items so a record for each item must be stored.

DevOps

1. GitHub Pipeline Flow Chart:

A visual representation of the GitHub Actions pipeline is depicted in the attached flow chart. This diagram outlines the sequential steps from setting up the environment to running tests, performing linting, assembling, and uploading artifacts. Each job in the pipeline has a distinct purpose, aligning with DevOps principles to streamline the development process.

2. Testing Output and Static Code Analysis:

Testing Output:

- The "Test Job" is dedicated to running unit tests using **./gradlew testDebugUnitTest**.
- Test reports are generated and uploaded as artifacts, ensuring visibility into the testing output.

Static Code Analysis:

- The "Lint Job" incorporates static code analysis using **./gradlew lintDebug**.
- Lint reports are uploaded as artifacts, providing insights into the code quality and adherence to coding standards.

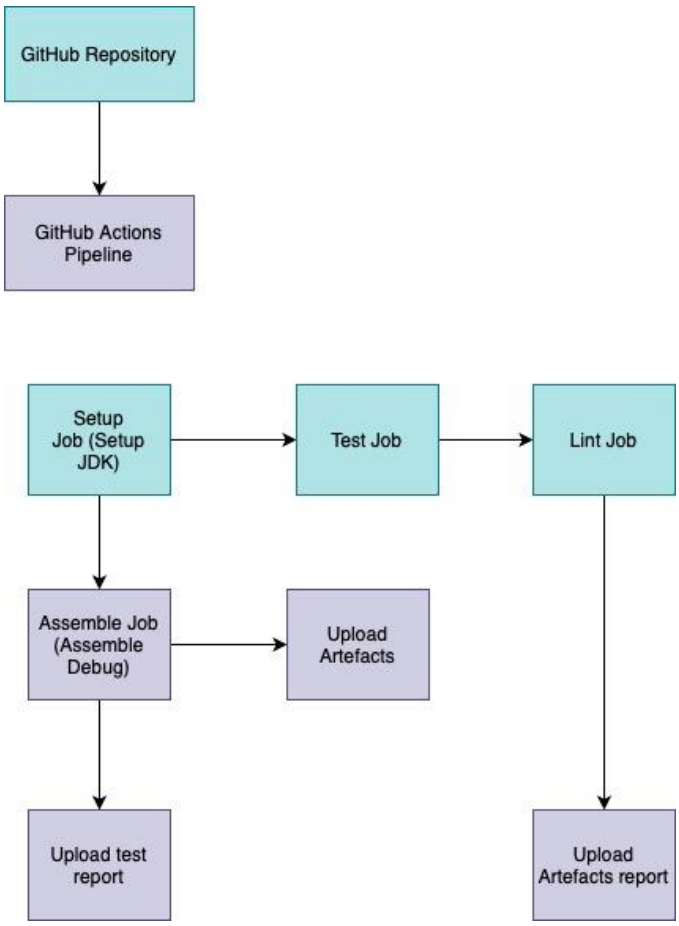

Figure 8 GitHub Pipeline

Running costs

This analysis unveils the expected monthly costs and growth projections for GitHub Actions and storage over the next two years. In alignment with a client-projected 10% annual growth, we explore the financial landscape of our development operations.

GitHub Free Plan Details:

- **Current Plan:**
  - GitHub Free
  - Unlimited public/private repos
  - 2,000 Actions minutes/month
  - 500MB of Packages storage
  - Community support
- **Current Usage:**
  - Actions: Ubuntu 2-core, Windows 2-core, macOS 3-core
  - Shared Storage: 0.09 GB-months
- **Next Payment Estimate:**
  - $0.00

GitHub Actions Monthly Cost Estimate:

- **GitHub Actions Minutes:**
  - Ubuntu 2-core: Included 12.00 minutes
  - Windows 2-core: Included 0.00 minutes
  - macOS 3-core: Included 0.00 minutes
- **Storage for Actions and Packages:**
  - Shared Storage: Included 0.09 GB-months
- **Billing Summary:**
  - Monthly Spending Limit: $0.00

GitHub Actions Monthly Cost Growth Projection:
Projected GitHub Actions Cost=$0.00Projected GitHub Actions Cost=$0.00

Storage Monthly Cost Growth Projection:
Projected Storage Cost=$0.00Projected Storage Cost=$0.00

Monthly Payment Estimate for Each Month Over Two Years:
GitHub Actions Monthly Payment Estimate=$0.00GitHub Actions Monthly Pay ment Estimate=$0.00

Change Management

Introducing our software product to Parktown Manufacturers involves a comprehensive change management strategy aimed at seamless adoption by both the organization and its users.

How and why will the organization adopt your software?
The organization's adoption hinges on aligning our software with Parktown Manufacturers' overarching objectives. We propose a phased implementation, starting with a pilot program that allows key stakeholders to witness the software's benefits firsthand. Emphasizing the positive impact on operational

efficiency, inventory management, and overall workplace organization will be integral to gaining organizational buy-in.

How and why will the users adopt your software?
User adoption is pivotal for the software's success. To encourage this, we prioritize user-friendliness and an intuitive interface. Training sessions and workshops will be conducted to familiarize users with the software's functionalities and emphasize its user-centric design. We will highlight the software's role in simplifying daily tasks, improving data accuracy, and providing real-time insights, thereby addressing pain points and showcasing tangible benefits.

What is your strategy to gain adoption from both the organization and the users?
Our strategy involves clear communication, education, and ongoing support. Regular communication channels, such as emails, and in-person sessions, will be utilized to keep both the organization and users informed about the software's implementation progress and benefits. Educational resources, including tutorials and documentation, will be provided to facilitate a smooth learning curve. Continuous feedback mechanisms and a responsive support system will address concerns promptly, ensuring a positive experience for both the organization and users. This holistic approach aims to create a culture of enthusiasm and confidence in embracing the new software.

Appendices

1. Declaration of Authenticity

Declaration of authenticity

I, Halalisani A Mdlalose ID Number, 0211145132083

hereby declare that this portfolio, and any evidence included therein, contains my own independent work and that I have not received help from other groups.

I confirm that we have not committed plagiarism in the accomplishment of this work, nor have I falsified and/or invented experimental data.

I accept the academic penalties that may be imposed for violations of the above.

HMdlalose

ST10092007

STUDENT SIGNATURE **DATE** 11/28/2023

© The Independent Institute of Education (Pty) Ltd 2023

Reference list

Bell, D. 2004. Explore the UML sequence diagram. 15 February 2004. [Online]. Available at: https://developer.ibm.com/articles/the-sequence-diagram/ [Accessed 06 October 2023].

bmc. 2020. *Data Storage Explained: Data Lake vs Warehouse vs Database,* 28 October 2020. [Online]. Available at: https://www.bmc.com/blogs/data-lake-vs-data-warehouse-vs-database-whats-the-difference/ [Accessed 29 September 2023]

Creately. 2022a. Sequence Diagram Tutorial-Complete Guide with Examples. 12 December 2022. [Online]. Available at: https://creately.com/guides/sequence-diagram-tutorial/ [Accessed 06 October 2023].

Creately. 2022b. UML Diagram Types Guide: Learn About All Types of UML Diagrams with Examples. 28 September 2022. [Online]. Available at: https://creately.com/blog/diagrams/uml-diagram-types-examples/ [Accessed 06 October 2023].

Guthrie, G. 2022. A simple guide to drawing your first state diagram (with examples). 10 March 2022. [Online]. Available at: https://nulab.com/learn/software-development/a-simple-guide-to-drawing-your-first-state-diagram-with-examples/ [Accessed 06 October 2023].

Lucidchart. (n.d.). *Entity-Relationship Diagram Symbols and Notation,* (n.d.). [Online]. Available at: https://www.lucidchart.com/pages/ER-diagram-symbols-and-meaning [Accessed 04 September 2023]

Plantuml. (n.d). Object Diagram (n.d). [Online]. Available at: https://plantuml.com/object-diagram [Accessed 05 October 2023].

Visual Paradigm. (n.d)a. What is Object Diagram?. (n.d). [Online]. Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-object-diagram/ [Accessed 05 October 2023].

Visual Paradigm. (n.d)b. What is Package Diagram?. (n.d). [Online]. Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-package-diagram/ [Accessed 05 October 2023].