ICT DEPARTMENT INVENTORY SYSTEM PRODUCT DOCUMENTATION

2024-01-05

Contents

1 Project Documentation

1.1 Team Memebers

- Name : NJIANGA Ayouba ___ Matricule : ICTU20222964 (Project Manager)
- Name: Ngane Emmanuel ___ Matricule :ICTU20222972 (Assistant Project Manager)
- Name: Yohmeh Marc Bian Bilingwe ___ Matricule: ICTU20223131
- Name:Mvogonka Christophe ulrich ___ Matricule: ICTU20222936
- Name: Fosang Che ___ Matricule: ICTU20222990
- Name: Nsaila Ridley.N ___ Matricule: ICTU20222976
- Name :Tanyu Ransom Tuwa ___ Matricule : ICTU20233584
- Name : SHUYE ANITA BERINYU ___ Matricule : ICTU2022318
- Name : Nyapem Joyce Patricia ___ Matricule : ICTU20223516
- Name : Ngankam Fotsing Ryan ___ Matricule : ICTU20223412

1.2 Introduction

The purpose of this document is to outline the requirements for developing an inventory system for our school. The system aims to facilitate the efficient and effective management of assets, resources, and infrastructure used by staff and students, such as equipment, supplies, books, and furniture.

In this document, headings are denoted by bold text and underlining, while subheadings are denoted by bold text alone. This convention helps to clearly distinguish different sections and subsections, ensuring ease of navigation and understanding.

The intended audience for this document consists of individuals directly involved in the development of the inventory system. It provides a comprehensive overview of the project's scope, objectives, deliverable, constraints, assumptions, and references.

1.2.1 Background

The ICT University currently has an inventory system that is divided into two categories: fixed assets and office supplies. However, the focus of interest lies in the fixed assets. In the ICT department, we utilize various equipment such as projectors, HDMI cables, sockets, monitor screens, RJ45 cables, and more. The Finance office conducts an annual inventory of the equipment upon arrival, adding it to the existing stock and assigning a cost to each item. Over time, the value of these assets decreases. When the equipment is used, it is removed from the system.

Each asset or equipment is assigned to a specific user, such as students and lecturers who utilize HDMI cables and projectors during their classes. As each asset has an assigned user, it is essential to track the location of the equipment when it is taken. For instance, when a student borrows equipment from the CISCO lab, they are required to provide their school ID or personal ID so that the university can identify the student responsible for the borrowed item. However, the current system lacks the ability to track the location of these items as they are moved, and there is no identification number assigned to each item. Consequently, the university must rely on manual processes to determine the availability of materials.

To enhance the system, several improvements are necessary. Firstly, each item should be assigned a unique identification number. Additionally, the system should store both the normal location of an item and its current location when it has been borrowed. When an individual borrows an item, they should provide an ID and sign it out. Upon returning the item, they sign it back in. By implementing these changes, the system will be capable of tracking the movement of items, calculating depreciation values, determining the quantity of faulty items, and providing an accurate count of items remaining in stock.

The inventory system currently operates on a Windows machine and is used locally within the university.

1.2.2 Objectives

1. Implement an identification numbering system: Assign unique identification numbers to each item in the inventory to facilitate tracking and management.

2. Track item locations: Develop a functionality that allows the system to record and update the normal location of each item as well as its current location when it is borrowed.

3. Enable user accountability: Require users to provide identification (such as a school ID or personal ID) when borrowing equipment, ensuring responsibility and accountability for borrowed items.

4. Automate check-in and check-out processes: Implement a system that allows users to sign items out when borrowed and sign them back in upon return, enabling efficient tracking of item movements.

5. Calculate depreciation value: Incorporate functionality to track the depreciation of assets over time, providing insight into the value and condition of equipment.

6. Maintain accurate inventory records: Ensure that the system maintains up-to-date and accurate records of available stock, including the quantity of faulty items and the remaining quantity in stock.
7. Improve reporting capabilities: Generate comprehensive reports on inventory levels, asset usage, and trends to facilitate informed decision-making and resource planning.
8. Enhance system usability: Develop a user-friendly interface that is intuitive and easy to navigate, facilitating efficient access to inventory information.
9. Local system deployment: Optimize the inventory system to run on Windows machines and operate locally within the ICT University.

These objectives aim to address the limitations of the current system and improve the management and tracking of assets and resources within the ICT University.

1.2.3 Scope

1.2.3.1 Objective The objective of this project is to develop an inventory system for the ICT University that facilitates the efficient and effective management of assets, resources, and infrastructure used by staff and students. The system will focus on tracking fixed assets, such as equipment used in the ICT department.

1.2.3.2 Features and Functionalities The inventory system will include the following features and functionalities: - Identification Numbering System: Assign unique identification numbers to each item in the inventory. - Item Location Tracking: Record and update the normal location and current location of items. - User Accountability: Require users to provide identification when borrowing equipment. - Check-in and Check-out Automation: Enable users to sign items out and back in. - Depreciation Tracking: Track the depreciation value of assets over time. - Accurate Inventory Records: Maintain up-to-date and accurate records of stock and faulty items. - Reporting Capabilities: Generate comprehensive reports on inventory levels, asset usage, and trends. - User-Friendly Interface: Develop an intuitive and easy-to-navigate interface. - Local System Deployment: Optimize the system to run on Windows machines and operate locally within the ICT University.

1.2.3.3 Exclusions The following items are excluded from the project scope: - Office supplies and other non-fixed assets. - Integration with external systems or databases. - Mobile or web-based application development.

1.2.3.4 Constraints The project must adhere to the following constraints: - Use of Windows machines for system deployment. - Compliance with relevant laws, regulations, policies, and standards. - Availability of resources, including hardware and software.

1.2.3.5 Assumptions The following assumptions are made for the project: - Sufficient cooperation and participation from stakeholders involved in the development and implementation of the inventory system. - Adequate availability of necessary data and information for inventory management.

1.2.3.6 Deliverable The project will deliver the following: - Developed inventory system with the specified features and functionalities. - User documentation and training materials. - Completed testing and quality assurance reports.

1.2.3.7 References The project will refer to the following resources: - Existing inventory management systems. - Relevant documentation and policies related to asset management.

1.2.4 Stakeholders

The key stakeholders involved in the project include: - **ICT University Administration**: Responsible for providing requirements and guidance, and ensuring alignment with university goals and policies. - **ICT Department**: Provides domain expertise, user requirements, and feedback during development and testing. - **Faculty and Staff**: End users of the system, responsible for asset management and utilizing the inventory system. - **Students**: Users who may borrow equipment from the inventory system for academic purposes. - **System Developers**: Responsible for system design, development, and implementation. - **Quality Assurance Team**: Ensures system functionality, usability, and adherence to specifications.

1.3 Planning

1.3.1 Project Plan

1.3.1.1 Project Initiation Phase

- Define project objectives, scope, and deliverable.
- Identify key stakeholders and establish communication channels.
- Conduct initial meetings with stakeholders to gather requirements and expectations.
- Develop a project charter outlining project goals, timelines, and resources.

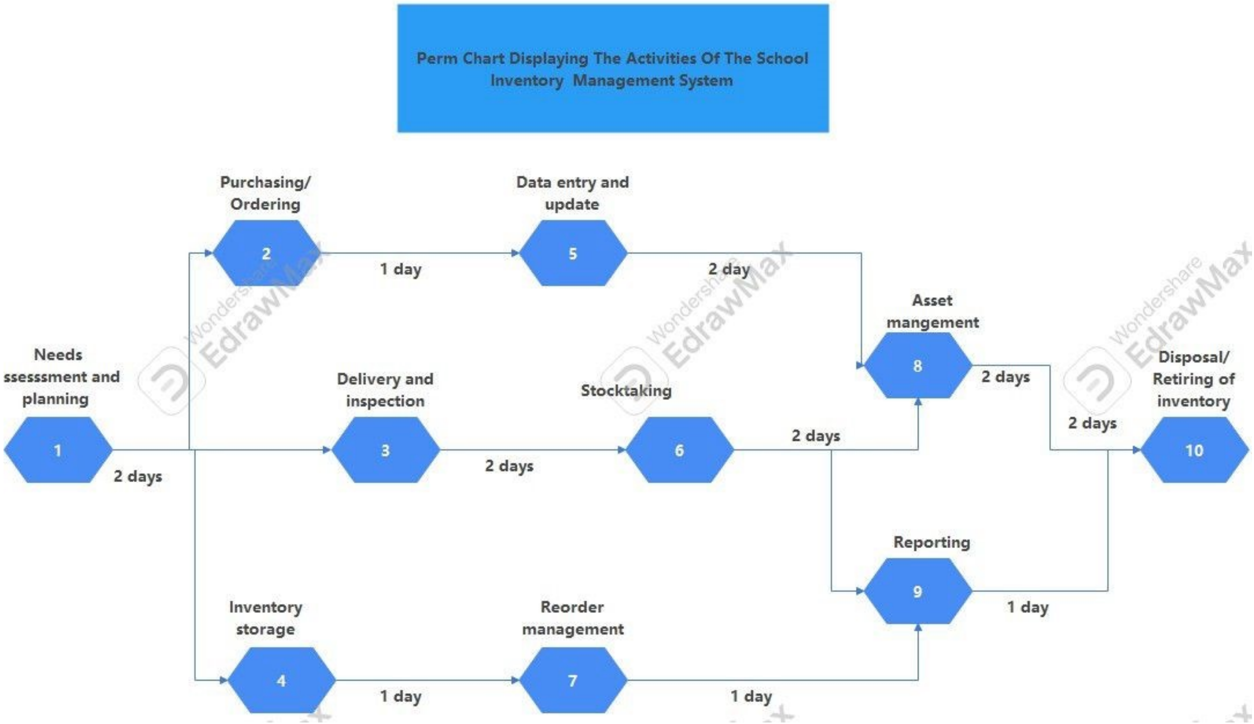


Figure 1: Perte chart

1.3.1.2 Requirements Gathering and Analysis Phase

- Conduct detailed interviews and workshops with stakeholders to gather system requirements.
- Analyze existing inventory management processes and identify pain points and areas for improvement.
- Document functional and non-functional requirements for the inventory system.
- Validate requirements with stakeholders and obtain sign-off.

1.3.1.3 System Design and Development Phase

- Develop a high-level system architecture and design based on the gathered requirements.
- Break down the system design into modules and components.
- Implement the identified features and functionalities of the inventory system.
- Conduct regular code reviews and testing to ensure quality and adherence to requirements.
- Iteratively refine and enhance the system based on feedback from stakeholders.

1.3.1.4 User Documentation and Training Phase

- Develop user documentation, including user manuals and guides.
- Create training materials and conduct training sessions for the system users.
- Ensure that users are familiar with the functionalities of the inventory system.
- Provide ongoing support and address user queries and concerns.

1.3.1.5 Testing and Quality Assurance Phase

- Develop a comprehensive test plan covering unit testing, integration testing, and system testing.
- Execute test cases to validate the system against the defined requirements.
- Identify and address any bugs, issues, or performance bottlenecks.
- Conduct user acceptance testing (UAT) with stakeholders to ensure system usability and functionality.
- Obtain sign-off on the tested and validated system.

1.3.1.6 Deployment and Roll-out Phase

- Prepare the system for deployment on Windows machines within the ICT University.
- Coordinate with the IT support team for system installation, configuration, and setup.
- Conduct data migration, if applicable, from any existing inventory systems.
- Ensure proper backup and disaster recovery mechanisms are in place.
- Conduct user training and provide assistance during the initial system roll-out.

1.3.1.7 Project Closure Phase

- Conduct a final review and assessment of the project against the defined objectives.
- Obtain feedback from stakeholders and identify lessons learned for future projects.
- Prepare and deliver a project completion report.
- Hand over the system to the ICT University administration and relevant stakeholders.
- Celebrate the successful completion of the project!

1.3.2 Project Schedule

Phase	Start Date	End Date
Project Initiation Phase	2023-11-17	2023-11-20
Requirements Gathering and Analysis	2023-12-01	2023-12-05
System Design and Development	2023-12-10	2024-01-01
User Documentation and Training	2024-01-01	2024-01-02
Testing and Quality Assurance	2023-12-20	2024-01-02
Deployment and Roll out	2024-01-04	2024-02-05
Project Closure Phase	2024-01-02	2024-01-04

Phase	Start Date	End Date

1.3.3 Resource Allocation

Proper resource management is crucial for the successful execution of a project. Assigning the right resources to each phase ensures that the required skills and expertise are available when needed. Below is an overview of the key resources assigned to each phase of the project:

- **Project Initiation Phase**: The Project Manager and Assistant Project manager are assigned during this phase. The Project Manager oversees the overall project and ensures its alignment with the objectives and stakeholders' expectations. The Assistant project manager works closely with stakeholders to gather requirements and define project scope.
- **Requirements Gathering and Analysis**: Shuye Anita Berinyu takes the lead in this phase, along with Mvogonka Christophe Ulrich, and Fosang che. They provide domain-specific knowledge and insights to ensure accurate requirement gathering and analysis.
- **System Design and Development**: The System Architect Ngane Emmanuel, provides expertise in designing the system architecture, while the Developers Yohmeh Marc Brian Bilingwe,Nsaila Ridely, and Ngane Emmanuel, are responsible for implementing the identified features and functionalities of the inventory system.
- **User Documentation and Training**: The Technical Writers Shuye Anita Berinyu, Tanyu Ransom Tuwa, Ngane Emmanuel, Njianga Ayouba, Joyce Patricia, Mvogonka Christophe, and Fosang Che create user documentation, including user manuals and guides.
- **Testing and Quality Assurance**: Testers and Quality Assurance Analysts play a vital role in this phase. They develop and execute test cases, identify bugs and issues, and ensure the system meets the defined requirements and quality standards. The entire team participated in the manual user testing, while the designer and developers carried out unit and integration testing.
- **Project Closure Phase**: The Project Manager leads the project closure phase, working closely with stakeholders to conduct a final review and assessment. The involvement of stakeholders is crucial for obtaining feedback and identifying lessons learned for future projects.

By effectively managing these resources and ensuring their availability and allocation according to the project schedule, the project team can work efficiently and deliver a successful outcome.

1.3.4 Risk Management

Risk management is a critical aspect of project planning and execution. Identifying and addressing potential risks in a proactive manner helps minimize their impact on the project's success. The following is an overview of the key steps involved in risk management:

1. **Risk Identification**: During this phase, the project team conducts a thorough analysis to identify potential risks that may arise throughout the project lifecycle. Risks can include technical challenges, resource constraints, scope changes, external dependencies, and more.
2. **Risk Assessment**: Once risks are identified, they are assessed based on their probability of occurrence and potential impact on the project. This evaluation helps prioritize risks and allocate appropriate resources for mitigation.

3. **Risk Mitigation Planning**: For high-priority risks, mitigation plans are developed to minimize their impact or likelihood of occurrence. These plans outline specific actions, responsibilities, and timelines for risk mitigation.
4. **Risk Monitoring and Control**: Risks are continuously monitored throughout the project. Regular status updates and progress reports are shared with stakeholders to keep them informed. If new risks emerge or existing risks evolve, they are addressed promptly to prevent any adverse effects on the project.
5. **Risk Response Execution**: When a risk materializes, the predefined mitigation plan is executed. This may involve allocating additional resources, adjusting project timelines, or taking alternative actions to minimize the impact of the risk.
6. **Risk Documentation and Communication**: Risks, mitigation plans, and their outcomes are documented to maintain a comprehensive record. This documentation serves as a reference for future projects and contributes to organizational learning. Regular communication with stakeholders ensures they are aware of the risks and the steps being taken to manage them.
7. **Risk Review and Lessons Learned**: At the end of the project, a comprehensive review of risks and their management is conducted. Lessons learned are documented, focusing on successful risk mitigation strategies and areas for improvement in future projects.

By incorporating a systematic approach to risk management, the project team can effectively identify, assess, and mitigate risks, thereby increasing the chances of project success and minimizing potential disruptions.

1.4 Requirements Analysis

1.4.1 User Requirements

1. **Administrators:** Administrators should be able to:
   - Add, remove, and manage user accounts.
   - Configure system settings and permissions.
   - Generate reports on inventory usage and system performance.
   - Perform data backups and restoration.
   - Integrate the inventory system with other school management systems.
   - Manage local stores and their inventory.
2. **Teachers and Staff:** Teachers and staff members should be able to:
   - Check out items from the inventory for classroom use.
   - Return borrowed items to the inventory.
   - Search for specific items based on various criteria.
   - Receive notifications for overdue items or low inventory levels.
   - View their borrowing history and transaction details.
   - Generate reports on their own inventory usage.
3. **Students:** Students should be able to:
   - Borrow items from the inventory for academic or extracurricular activities.
   - Return borrowed items to the inventory.
   - Receive notifications for overdue items.
   - View their borrowing history and transaction details.
4. **Retailers:** Retailers who supply inventory items to the school should be able to:
   - Provide updated inventory information to the school.
   - View reports on inventory usage and payment history.

1.4.2 Functional Requirements

1. **Inventory Management:** The system should be able to track all items in the inventory, including their current status (e.g., checked out, available, under maintenance) and location.
2. **Transaction Management:** The system should support the insertion, modification, deletion, cancellation, and error checking of transactions.
3. **User Management:** The system should allow administrators to add, remove, and manage user accounts.
4. **Reporting:** The system should generate reports on inventory usage, including checkouts, returns, and overdue items.
5. **Search Functionality:** Users should be able to search the inventory based on various criteria, such as item name, category, or location.
6. **Notifications:** The system should send notifications for important events, such as overdue items or low inventory levels.
7. **Integration:** The system should integrate with other systems used by the school, such as the school management system.
8. **Security:** The system should have authentication and authorization mechanisms to ensure that only authorized users can access the system and perform certain actions.
9. **Data Backup and Restoration:** The system should have mechanisms for backing up and restoring data.
10. **Local Store Management:** The system should be able to manage local stores.
11. **Retailer's Privacy:** The system should ensure the privacy of the retailers.

1.4.3 Non-Functional Requirements

1. **Security:** The system should provide secure access control mechanisms to protect sensitive inventory data.
2. **Usability:** The system should have a user-friendly interface that is intuitive and easy to navigate.
3. **Performance:** The system should be able to handle a large number of inventory items and transactions efficiently without significant delays or downtime.
4. **Reliability:** The system should be reliable and available for users to access at all times, with minimal disruptions or system failures.
5. **Scalability:** The system should be able to accommodate the growing inventory needs of the school and handle increased user and data load.
6. **Compatibility:** The system should be compatible with different devices, operating systems, and web browsers.
7. **Data Accuracy:** The system should ensure accurate and up-to-date inventory data through proper validation and error checking mechanisms.
8. **Data Integrity:** The system should maintain the integrity of inventory data by preventing unauthorized modifications or deletions.
9. **Data Privacy:** The system should comply with data privacy regulations and protect the privacy of users' personal information.
10. **Data Backup and Recovery:** The system should have regular backup procedures in place to prevent data loss and be able to recover data in case of system failures or disasters.

1.4.4 Use Cases

1. **Add Item to Inventory:** This use case involves adding a new item to the inventory system, including details such as item name, category, quantity, and location.
2. **Check Out Item:** This use case allows users to check out an item from the inventory, updating its status and associating it with the borrower's information.
3. **Return Item:** This use case enables users to return a borrowed item to the inventory, updating its status and removing the association with the borrower.

4. **Search Inventory:** This use case allows users to search for specific items in the inventory based on various criteria, such as item name, category, or location.
5. **Generate Inventory Report:** This use case involves generating reports on inventory usage, including checkouts, returns, and overdue items.
6. **Manage User Accounts:** This use case allows administrators to add, remove, and manage user accounts, including roles and permissions.
7. **Integrate with School Management System:** This use case involves integrating the inventory system with the school management system to share relevant data and ensure seamless information flow.
8. **Send Notifications:** This use case allows the system to send notifications to users for important events, such as overdue items or low inventory levels.
9. **Perform Data Backup and Restoration:** This use case involves performing regular data backups and being able to restore data in case of system failures or data loss.

1.4.5 Data Requirements

1. **Inventory Data:** The system should store information about each item in the inventory, including item ID, name, category, quantity, status, location, and any associated borrower information.
2. **User Data:** The system should store user account information, including username, password, role, and permissions.
3. **Transaction Data:** The system should record transaction details, such as the date and time of checkout or return, the item involved, and the borrower information.
4. **Notification Data:** The system should store notification preferences for users, including the events for which they want to receive notifications and their preferred contact method.
5. **Backup Data:** The system should create and store backup data on a regular basis to ensure data integrity and enable restoration in case of system failures or data loss.

1.5 System Design

1.5.1 Architecture Design

The architecture design of the ICT department inventory system is based on the VBA programming language within the Microsoft Excel environment. The system follows a client-server architecture, where the Excel workbook acts as the client application and interacts with a back-end database.

The client-side architecture utilizes VBA modules and forms within the Excel workbook to handle user interactions, data processing, and presentation. The VBA code is organized into modules based on functionality, following best practices for code modularity and maintainability.

On the server-side, the system utilizes an Excel worksheet as the database to store and manage the inventory data. The worksheet consists of multiple tables representing different entities in the system, such as inventory items, suppliers, and transactions. The relationships between these tables are defined using appropriate key fields.

1.5.2 Component Design

The component design of the ICT department inventory system involves breaking down the system's functionality into smaller, manageable components. Each component is responsible for specific tasks, such as data validation, data retrieval, data storage, and reporting.

The VBA code is organized into reusable functions and subroutines that can be called from different parts of the system. This modular approach promotes code re-usability, maintainability, and ease of testing.

Some key components of the system include:

1. Data Validation Component: This component handles the validation of user input, ensuring that only valid data is entered into the system. It includes functions to validate data formats, check for duplicate entries, and enforce business rules.
2. Data Retrieval Component: This component is responsible for retrieving data from the Excel worksheet database based on user queries. It includes functions to search for inventory items, generate reports, and provide data for analysis.
3. Data Storage Component: This component handles the storage and management of inventory data in the Excel worksheet database. It includes functions to insert, update, and delete data records as well as manage data relationships.
4. Reporting Component: This component generates various reports based on user requirements. It includes functions to generate inventory lists, transaction histories, and statistical summaries.

1.5.3 Database Design

The database design of the ICT department inventory system involves defining the structure and relationships of the Excel worksheet tables to efficiently store and retrieve data. The specific design considerations for the database design include:

1. **Table Design**: Based on the system's requirements, the Excel worksheet is structured into multiple tables to represent different entities. Each table has appropriate columns to store the relevant data, following normalization principles to avoid data redundancy and ensure data integrity.
2. **Data Relationships**: The relationships between the Excel worksheet tables are defined using appropriate key fields. This enables the system to establish connections between related data and perform efficient data retrieval operations.
3. **Data Validation**: Data validation rules are implemented within the VBA code to ensure data integrity and consistency. This may include enforcing data types, validating input ranges, and checking for referential integrity between tables.

1.5.4 User Interface Design

The user interface design of the ICT department inventory system focuses on providing an intuitive and user-friendly interface for users to interact with the system. The user interface is developed within the Excel workbook using VBA forms, controls, and worksheets.

Key considerations for the user interface design include:

1. Form Design: Forms are designed to capture user input, display data, and facilitate user interactions. The form layout is organized logically, with clear labels, input fields, and buttons. Validation mechanisms are implemented to guide users in entering valid data.
2. Navigation Design: The user interface includes navigation controls to allow users to move between different forms and functionalities. Clear and intuitive navigation menus, buttons, or tabs are provided to enhance the user experience.
3. Reporting and Visualization: The user interface includes features to generate and display reports based on user queries. This may involve displaying data in tables, charts, or graphs to facilitate data analysis and decision-making.
4. Error Handling and Feedback: The user interface incorporates error handling mechanisms to provide meaningful error messages when users encounter issues. Feedback mechanisms, such as progress indicators or confirmation messages, are implemented to keep users informed about the system's status.

1.6 Implementation

The implementation phase of the ICT department inventory system involves translating the system design into working software. The software development process follows a systematic approach to ensure the successful realization of the system requirements.

1.6.1 Software Development

During the software development phase, the following steps are typically followed:

1. **Coding**: The VBA code is written based on the system design specifications. The code implements the functionality outlined in the design, including data validation, data retrieval, data storage, and reporting. Best practices for coding style, readability, and maintainability are followed to ensure high-quality code.

2. **Testing**: The developed code is thoroughly tested to identify and fix any bugs or issues. Different testing techniques, such as unit testing and integration testing, are employed to validate the functionality of individual components and their interactions within the system. Test cases are designed to cover a wide range of scenarios and edge cases to ensure the robustness of the system.

3. **Debugging**: If any issues or errors are identified during testing, debugging techniques are used to identify the root cause and rectify the problems. Debugging tools and techniques provided by the VBA development environment are utilized to step through the code, inspect variables, and analyze the program flow.

4. **Optimization**: The code is optimized for performance, aiming to enhance the speed and efficiency of the system. This may involve identifying and eliminating bottlenecks, optimizing database queries, and improving algorithmic efficiency. Profiling tools and techniques are used to measure and analyze the system's performance, making necessary adjustments to improve its overall speed and responsiveness.

1.6.2 Coding Guidelines

The implementation phase of the ICT department inventory system involves translating the system design into working software. The software development process follows a systematic approach to ensure the successful realization of the system requirements. Here are some common coding guidelines followed during the implementation of the ICT department inventory system:

1. **Naming Conventions**: Descriptive and meaningful names are used for variables, functions, and modules. Camel case or snake case naming conventions are typically followed to improve code readability.

2. **Indentation and Formatting**: Consistent indentation and formatting are applied to improve code readability. Indentation levels are typically set to four spaces or one tab, and code blocks are appropriately organized.

3. **Commenting**: Code is well-documented using comments to explain its purpose, functionality, and any complex algorithms or business rules. Comments are written in a clear and concise manner to facilitate understanding and future maintenance.

4. **Modularity and Re-usability**: The code is organized into modular units, such as functions and subroutines, to promote re-usability and maintainability. Common functionalities are encapsulated into reusable modules, reducing code duplication and improving code manageability.

1.7 Testing

1.7.1 Login Functionality

1.7.1.1 Objective: The objective of the test is to ensure that login functionality brings the user to the main page or dashboard of the system which has all the menus of the system.

1.7.1.2 Features:

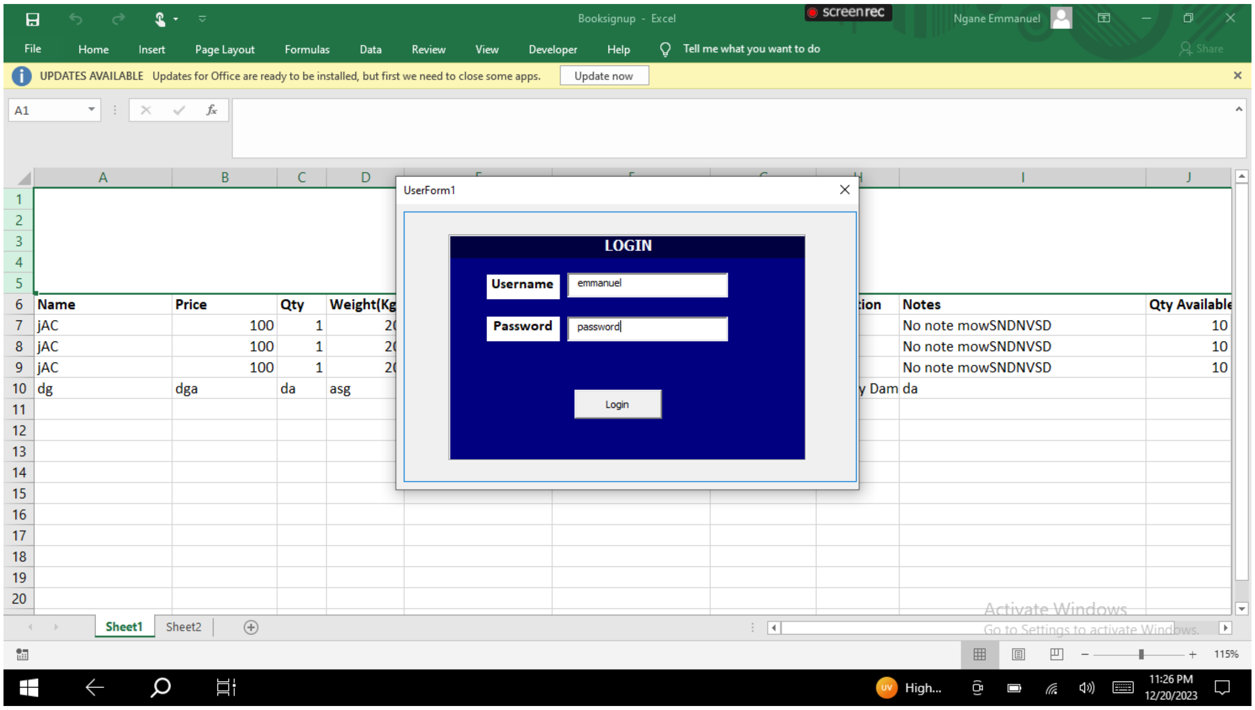
- clicking on the launch app button
- The login interface pops up

1.7.1.3 Test Scenarios:

- **Scenario 1**: Enter the valid credentials and click on the login button
- **Scenario 2**: Enter invalid data and click the login button

1.7.1.4 Test Steps:

- open workbook
- Verify login window shows properly



UPDATES AVAILABLE Updates for Office are ready to be installed, but first we need to close some apps. Update now

UserForm1

LOGIN

Username: emmanuel

Password: password

Login

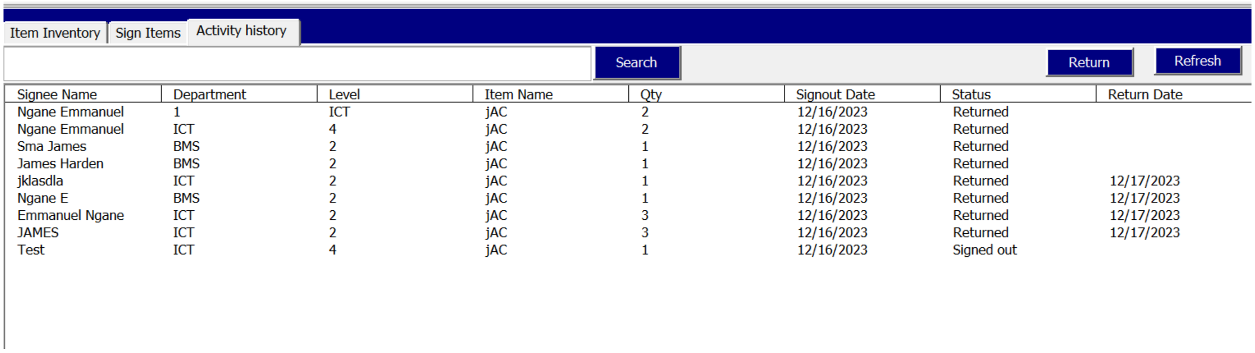
	A	B	C	D	H	I	J
6	Name	Price	Qty	Weight(Kg	tion	Notes	Qty Available
7	jAC	100	1	2(		No note mowSNDNVSD	10
8	jAC	100	1	2(		No note mowSNDNVSD	10
9	jAC	100	1	2(		No note mowSNDNVSD	10
10	dg	dga	da	asg	y Dam	da	

Figure 2: login window

- Enter the valid credentials in input boxes provided
- click on the login button
- Verify that data the window changes to the main-view window
- Enter invalid data and click the login button
- verify that the proper error message was displayed

1.7.1.5 Expected Results:

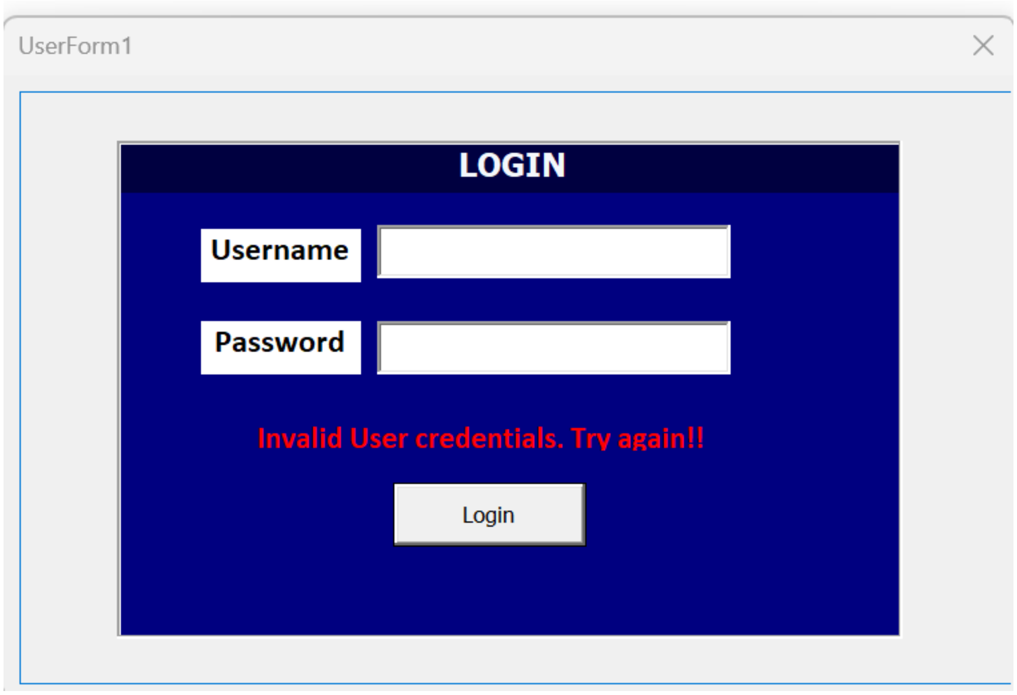
- **Successful login**: Entering the right credentials brings you to the home-page/Main-view of the system as shown below



Signee Name	Department	Level	Item Name	Qty	Signout Date	Status	Return Date
Ngane Emmanuel	1	ICT	jAC	2	12/16/2023	Returned	
Ngane Emmanuel	ICT	4	jAC	2	12/16/2023	Returned	
Sma James	BMS	2	jAC	1	12/16/2023	Returned	
James Harden	BMS	2	jAC	1	12/16/2023	Returned	
jklasdla	ICT	2	jAC	1	12/16/2023	Returned	12/17/2023
Ngane E	BMS	2	jAC	1	12/16/2023	Returned	12/17/2023
Emmanuel Ngane	ICT	2	jAC	3	12/16/2023	Returned	12/17/2023
JAMES	ICT	2	jAC	3	12/16/2023	Returned	12/17/2023
Test	ICT	4	jAC	1	12/16/2023	Signed out	

Figure 3: inventory system main-view

- Entering wrong credentials or leaving it blank will display an error messge shown below



Invalid User credentials. Try again!!

Figure 4: error handling message

1.7.2 Add items functionality

1.7.2.1 Sub-unit 1: Testing Pop-up Display

1.7.2.1.1 Objective: The objective of this sub-unit is to verify that the pop up is displayed correctly when add button is clicked

1.7.2.1.2 Features:

- Click on the add button
- Observe menu displayed

1.7.2.1.3 Test Scenarios:

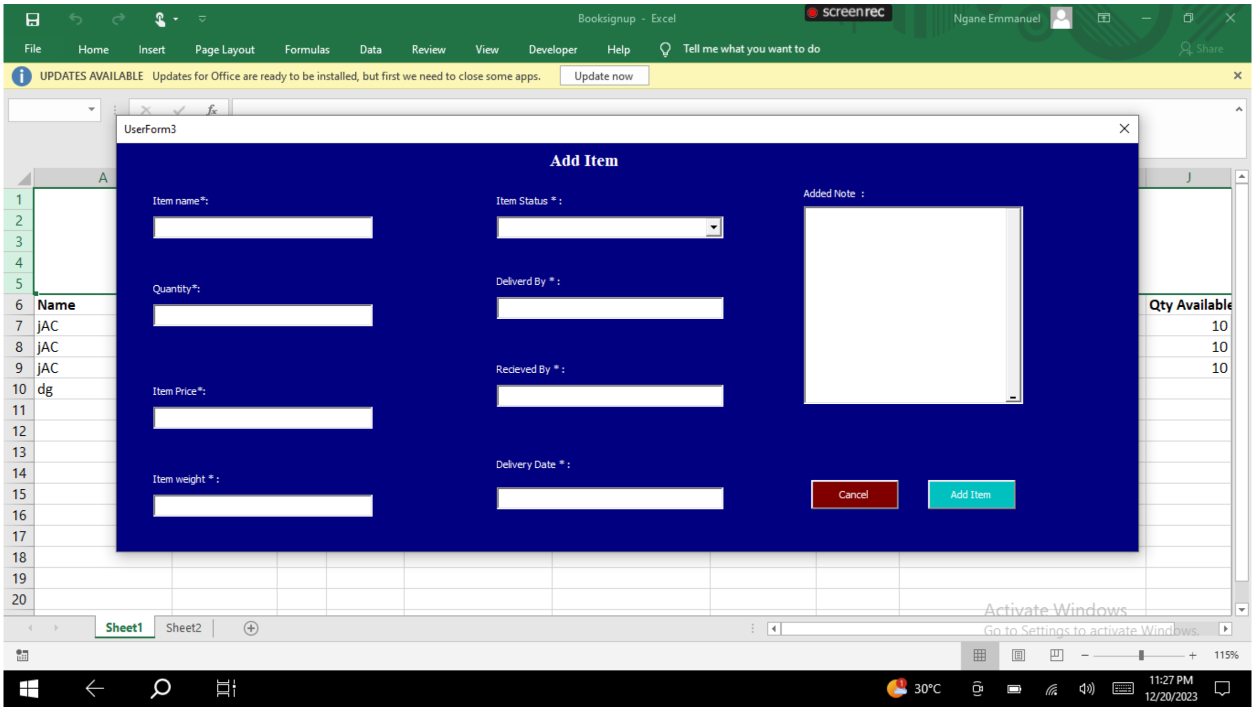
- **Scenario 1**: Click the add button

1.7.2.1.4 Test Steps:

- Click the add button
- Observe the proper display of an add items window

1.7.2.1.5 Expected Results:

- Successful display of the add item window with all its content



Name	Qty Available
jAC	10
jAC	10
jAC	10
dg	

Figure 5: Add item window

1.7.2.2 Sub-unit 2: Adding of item.

1.7.2.2.1 Objective: The objective of this sub-unit is to verify if items are added successfully into Sheet1 after clicking the add button on the add items window.

1.7.2.2.2 Features:

- Entering valid information in the add items window fields.
- Clicking the "add" button triggers the saving of information to "Sheet1."
- Appropriate validation and error handling mechanisms are in place.

1.7.2.2.3 Test Scenarios:

- **Scenario 1**:Enter valid information in the add item window fields and click the "add" button. Verify that the information is saved to "Sheet1" correctly.
- **Scenario 2**: Enter invalid or incomplete information and attempt to add.. Verify that appropriate error messages are displayed, and the information is not saved.

1.7.2.2.4 Test Steps:

- Enter information in the add item window
- click the "add" button
- Verify that a new row has been added in sheet1
- Verify the saved information in "Sheet1." is the last row
- Verify the information is correctly labeled as specified in the add item window

1.7.2.2.5 Expected Results:

- Adding item with all fields fills correctly adds a new row of information in sheet1, a success message is displayed, window is closed and user returned to the main-view
- Adding items with missing data displays an error to fill all required fields

1.7.3 Edit items functionality

1.7.3.1 Sub-unit 1: Testing Pop-up Display

1.7.3.1.1 Objective: The objective of this sub unit is to verify that the Edit item window is displayed correctly when edit button is clicked

1.7.3.1.2 Features:

- Click on the edit button
- Observe edit window displayed

1.7.3.1.3 Test Scenarios:

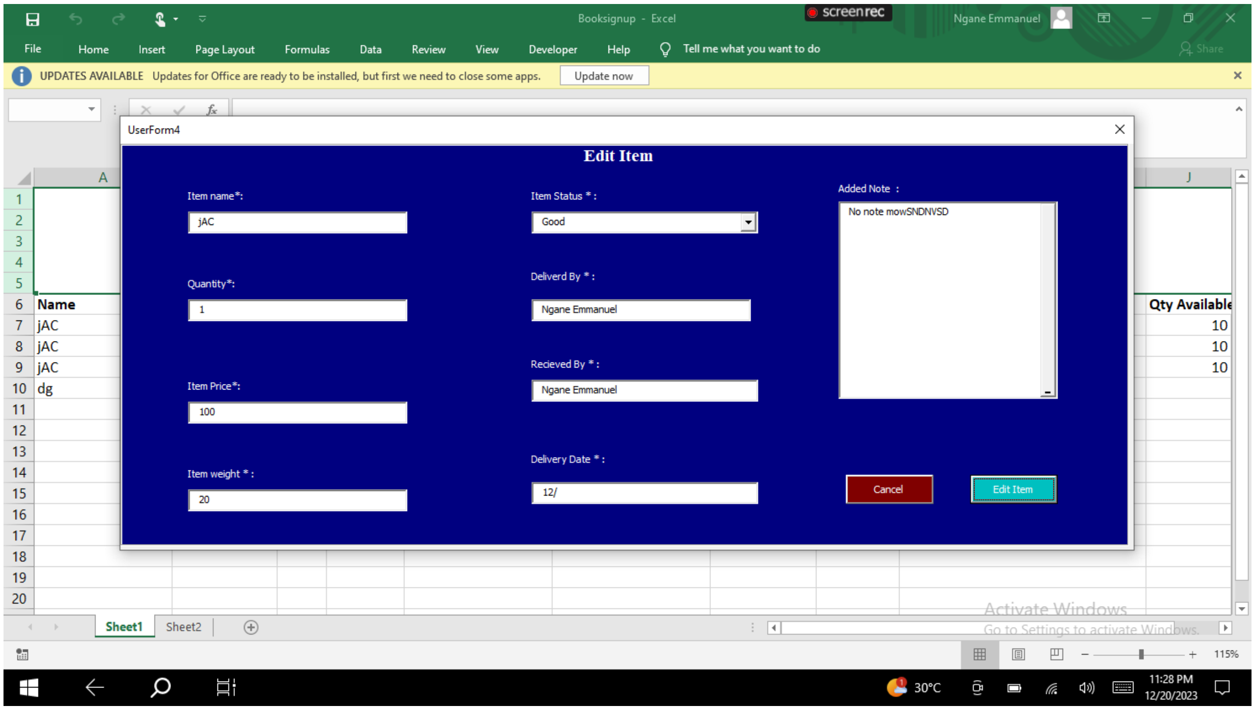
- **Scenario 1**: Click the edit button

1.7.3.1.4 Test Steps:

- Click the edit button
- Observe the proper display of an edit items window
- Observe if the input fields are pre-filled

1.7.3.1.5 Expected Results:

- Successful display of the edit item window with all its content
- Successful; pre-filling of the input boxes.


Booksignup - Excel
File
Home
Insert
Page Layout
Formulas
Data
Review
View
Developer
Help
Tell me what you want to do
UPDATES AVAILABLE Updates for Office are ready to be installed, but first we need to close some apps.
Update now
UserForm4
Cancel
Edit Item

Edit Item

Item name*: jAC

Item Status * : Good

Added Note : No note mowSNDNVSD

Quantity*: 1

Deliverd By * : Ngane Emmanuel

Recieved By * : Ngane Emmanuel

Item Price*: 100

Item weight * : 20

Delivery Date * : 12/

	Name	Qty Available
6	Name	Qty Available
7	jAC	10
8	jAC	10
9	jAC	10
10	dg	

Figure 6: Edit menue

1.7.3.2 Sub-unit 2: Editing of item.

1.7.3.2.1 Objective: The objective of this sub-unit is to verify if items are updated successfully into Sheet1 after clicking the "Edit" button on the edit items window.

1.7.3.2.2 Features:

- Entering valid information in the edit items window fields.
- Clicking the "Edit" button triggers the updating of information in "Sheet1."
- Appropriate validation and error handling mechanisms are in place.

1.7.3.2.3 Test Scenarios:

- **Scenario 1**:Enter valid information in the edit item window fields and click the "Edit" button. Verify that the information is updated in "Sheet1" correctly.
- **Scenario 2**: Enter invalid or incomplete information and attempt to edit. Verify that appropriate error messages are displayed, and the information is not updated.

1.7.3.2.4 Test Steps:

- Enter information in the edit item window
- click the “Edit” button
- Verify that a the information for item selected has been updated in sheet1
- Verify the information is correctly labeled as specified in the add item window

1.7.3.2.5 Expected Results:

- Editing item with all fields fills correctly updates the information of the selected item in sheet1, a success message is displayed, window is closed and user returned to the main-view
- Editing items with missing data displays an error to fill all required fields

1.7.4 Delete items functionality

1.7.4.1 Objective: The objective of this test s to verify that any item or record can be deleted from the database.

1.7.4.2 Features:

- Selecting an item allows you to use the delete option

1.7.4.3 Test Conditions:

1.7.4.4 Test Scenarios:

- **Scenario 1**: Select an items to delete, try to click delete
- **Scenario 2**: Don’t select an item and try to click delete

1.7.4.5 Test Steps:

- Select an item on the main-view you wish to delete
- Click the delete button
- Refresh the view
- Try deleting without selecting an item

1.7.4.6 Expected Results:

- On selecting items, you are able to access the delete button and the item is removed from sheet1
- When item is not selected,you are unable to acces the delete button

1.7.4.7 Test Data Data in sheet1

1.7.5 Refresh functionality

1.7.5.1 Objective: The objective of the test is to verify that the “refresh” functionality correctly updates the main view with the most recent data from “Sheet1.”

1.7.5.2 Features:

- Retrieving data from “Sheet1”
- Updating the main view with the retrieved data
- Handling any errors or exceptions during the refresh process

1.7.5.3 Test Conditions:

- A populated “Sheet1”: Test the functionality when “Sheet1” contains data that needs to be refreshed in the main view.
- Empty “Sheet1”: Test the functionality when “Sheet1” is empty and ensure that the main view handles this condition appropriately.

1.7.5.4 Test Scenarios:

- **Scenario 1**: Click the refresh button when “Sheet1” contains new data. Verify that the main view is updated with the most recent data.
- **Scenario 2**: Click the refresh button when “Sheet1” is empty. Verify that the main view handles this condition gracefully, such as displaying a message indicating no data is available.

1.7.5.5 Test Steps:

- Click the refresh button,
- Validate the updated data in the main view by crosschecking the data in “sheet1” and “sheet2” with the respective data in the Listbox1 and ListBox2, and ListBox3. The data in “sheet1” is checked agains the information in “ListBox1” and “ListBox2” and that of “sheet2” against “ListBox3”.

1.7.5.6 Expected Results: The information in all three ListBoxes(1, 2, 3) in the main view reflects the updated data from “Sheet1” and “sheet2” as described in the steps above accurately.

1.7.5.7 Test Data: Data in “sheet1” and “sheet2”.

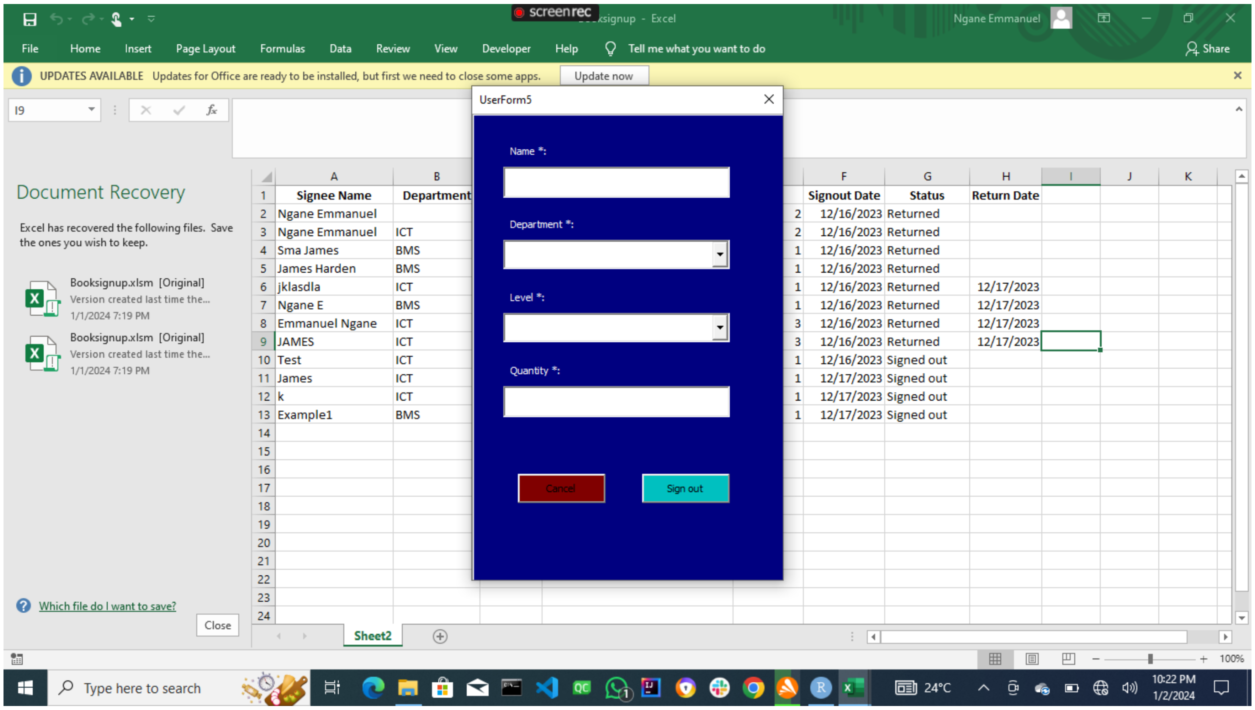
1.7.6 sign out item functionality

1.7.6.1 Sub-unit 1: Testing Pop-up Display

1.7.6.1.1 Objective: The objective of this sub-unit is to verify that the pop-up is displayed correctly when an item is selected.

1.7.6.1.2 Features:

- Selecting an item triggers the display of the pop-up.



UserForm5
Name *:
Department *:
Level *:
Quantity *:
Cancel
Sign out
Document Recovery
Excel has recovered the following files. Save the ones you wish to keep.
Booksignup.xlsm [Original]
Version created last time the...
1/1/2024 7:19 PM
Booksignup.xlsm [Original]
Version created last time the...
1/1/2024 7:19 PM
Which file do I want to save?
Close
UPDATES AVAILABLE Updates for Office are ready to be installed, but first we need to close some apps.
Update now
Sheet2

	Signee Name	Department		Signout Date	Status	Return Date
2	Ngane Emmanuel		2	12/16/2023	Returned	
3	Ngane Emmanuel	ICT	2	12/16/2023	Returned	
4	Sma James	BMS	1	12/16/2023	Returned	
5	James Harden	BMS	1	12/16/2023	Returned	
6	jklasdla	ICT	1	12/16/2023	Returned	12/17/2023
7	Ngane E	BMS	1	12/16/2023	Returned	12/17/2023
8	Emmanuel Ngane	ICT	3	12/16/2023	Returned	12/17/2023
9	JAMES	ICT	3	12/16/2023	Returned	12/17/2023
10	Test	ICT	1	12/16/2023	Signed out	
11	James	ICT	1	12/17/2023	Signed out	
12	k	ICT	1	12/17/2023	Signed out	
13	Example1	BMS	1	12/17/2023	Signed out	

* The pop-up contains the required input fields and buttons. * The pop-up is positioned correctly and appears as expected on the screen.

1.7.6.1.3 Test Scenarios:

- **Scenario 1**: Select an item and verify that the pop-up is displayed with the necessary input fields and buttons.
- **Scenario 2**: Select another item and ensure that the previous pop-up is closed and a new pop-up is displayed.

1.7.6.1.4 Test Steps:

- Select an item.
- Observe the pop-up.
- Verify its appearance and contents.

1.7.6.1.5 Expected Results:

- Verifying that the pop-up is displayed and the main window is hidden.
- contains input fields including Name, Department, Level, and Quantity.
- Contains two button at the bottom Cancel and Sign out.

1.7.6.2 Sub-unit 2: Testing Signing Out Process

1.7.6.2.1 Objective: The objective of this sub-unit is to verify the signing out of items process, including entering information, clicking the “sign out” button, and saving the information to “Sheet2.”

1.7.6.2.2 Features:

- Entering valid information in the pop-up fields.
- Clicking the “sign out” button triggers the saving of information to “Sheet2.”
- Appropriate validation and error handling mechanisms are in place.

1.7.6.2.3 Test Scenarios:

- **Scenario 1**: Enter valid information in the pop-up fields and click the “sign out” button. Verify that the information is saved to “Sheet2” correctly.
- **Scenario 2**: Enter invalid or incomplete information and attempt to sign out. Verify that appropriate error messages are displayed, and the information is not saved.

1.7.6.2.4 Test Steps:

- Select an item,
- Enter information in the pop-up
- click the “sign out” button
- Verify that a new row has been added in sheet2
- Verify the saved information in “Sheet2.” is the last row
- Verify the information is correctly labeled as specified in the pop-up

1.7.6.2.5 Expected Results:

- The information is saved correctly to “Sheet2,”
- Error messages are displayed when all one or more required field are empty
- The signing out process behaves as expected based on the test conditions.

1.7.7 returning items functionality:

1.7.7.1 Objective: The objective of this test is to verify that the “returning items” functionality correctly updates the item’s status and quantity in both “Sheet1” and “Sheet2” after the return button is clicked.

1.7.7.2 Features:

- Selecting an item to be returned
- Clicking the return button
- Updating the item’s status to “returned”
- Updating the quantity returned in both “Sheet1” and “Sheet2”

1.7.7.3 Test Scenarios:

- **Scenario 1**: Select an item that has not been returned, click the return button, and verify that the item’s status is updated to “returned” in both “Sheet2” and “Available quantity” column is updated to be less than the original quantity in “Sheet1.”
- **Scenario 2**: Select an item that has already been returned, click the return button, and ensure that the item’s status remains unchanged in both “Sheet2” and “Available quantity” column also remain unchanged in “Sheet1.”

1.7.7.4 Test Steps:

- Select an item
- Click the return button
- Verify the updated status in “Sheet2” Quantity available in “Sheet1,”
- Validating the quantity returned.

1.7.7.5 Expected Results:

- item’s status is correctly updated to “returned” in “Sheet2”
- Quantity Available for item is correctly updated in “Sheet1,”

1.7.7.6 Test Data: Data in Sheet1 and Sheet2

1.8 Deployment

The deployment phase involves making the ICT department inventory system available for use by end-users. This phase includes installation, configuration, user training, and user acceptance testing.

1.8.1 Installation

The installation process consists of deploying the system on the target environment. The following steps are typically followed:

1. **System Requirements**: Ensure that the target environment meets the necessary system requirements, including hardware specifications, software dependencies, and network connectivity.
2. **Installation Package**: Prepare an installation package that includes the necessary files and resources required to set up the ICT department inventory system. This may involve creating an installer or packaging the relevant files for deployment.
3. **Installation Procedure**: Execute the installation package on the target environment following the provided instructions. This may involve running an installer program, copying files to the appropriate locations, configuring databases, and setting up any required services.

1.8.2 Configuration

Once the ICT department inventory system is installed, it needs to be configured according to the specific needs of the school. The configuration process typically involves the following steps:

1. **Database Configuration**: Set up and configure the database that will be used by the system. This includes configuring access permissions.
2. **System Settings**: Configure system-wide settings such as default values, system behavior, and access control. This may involve specifying the currency format, setting up email notifications, defining user roles and permissions, and configuring integration with other systems if applicable.
3. **Customization**: Customize the system to align with the school’s specific requirements. This may include configuring custom fields, creating custom reports, and modifying user interfaces to match the school’s branding.

1.8.3 User Training

User training is an essential part of deploying the ICT department inventory system. It ensures that end-users are familiar with the system's features and functionalities. The following steps are typically involved in user training:

1. **Training Materials**: Develop training materials such as user manuals, guides, or videos that provide step-by-step instructions on how to use the system. These materials should cover all aspects of the system, including data entry, data retrieval, reporting, and any specific workflows.

2. **Training Sessions**: Conduct training sessions for the end-users, either in a classroom setting or through online sessions. The training should be interactive, allowing users to practice using the system and ask questions. It is also helpful to provide hands-on exercises and real-world scenarios relevant to the school's inventory management processes.

3. **Documentation**: Provide documentation that summarizes the key concepts and procedures covered in the training sessions. This documentation can serve as a reference guide for users to refresh their knowledge or troubleshoot issues.

1.8.4 User Acceptance Testing

User acceptance testing (UAT) is performed to ensure that the ICT department inventory system meets the requirements and expectations of the end-users. The following steps are typically followed in UAT:

1. **Test Planning**: Plan the UAT process, including defining test cases, test scenarios, and acceptance criteria. Test cases should cover various aspects of the system, including different functionalities, workflows, and edge cases.

2. **Test Execution**: Execute the test cases by following the defined scenarios and acceptance criteria. End-users or designated testers perform the tests, recording any issues or discrepancies encountered during the testing process.

3. **Issue Resolution**: If any issues are identified during UAT, they are reported to the development team for resolution. The development team investigates and addresses the reported issues, ensuring that the system functions as expected.

4. **Sign-off**: Once all test cases have been executed, and any identified issues have been resolved, the end-users or stakeholders provide sign-off, indicating their acceptance of the system.

The deployment phase concludes when the ICT department inventory system has been successfully installed, configured, and tested by the end-users, and they are ready to start using it in their day-to-day operations.

1.9 User Manuals

1.9.1 Introduction

The ICT Department Inventory System is a comprehensive software solution designed specifically for the Information and Communication Technology (ICT) department within a school. It serves as a centralized platform for efficiently managing and tracking the inventory of technological equipment, software licenses, and other ICT-related resources used by the department.

This inventory system offers several key benefits to the ICT department. It enables better control and organization of ICT assets, ensuring that the department has an accurate and up-to-date record of the equipment available. By maintaining a clear overview of the inventory, the ICT department can optimize resource allocation, plan for upgrades, and ensure the smooth functioning of technology within the school.

With the ICT Department Inventory System, staff members can easily track the location, condition, and maintenance history of ICT assets. This streamlines the process of borrowing equipment, scheduling maintenance tasks, and identifying items that require repair or replacement. By automating these processes, the system increases efficiency and reduces the risk of equipment loss or misuse.

Furthermore, the ICT Department Inventory System promotes collaboration and communication among staff members. It enables seamless sharing of information about inventory availability, equipment reservations, and upcoming maintenance activities. This fosters better coordination within the department and ensures that resources are effectively utilized.

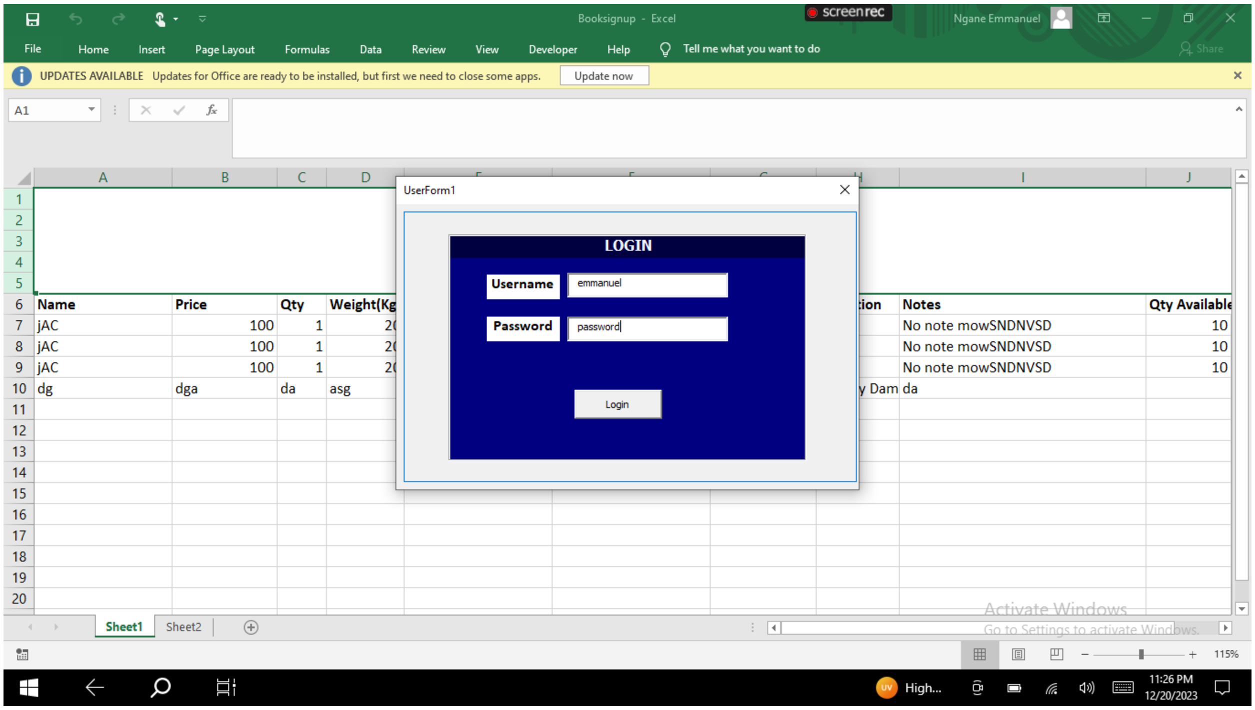
This user manual provides detailed instructions on how to utilize the ICT Department Inventory System effectively. It covers various functionalities, including user login, adding items to the inventory, editing and deleting items, signing out items to institution members, returning items, generating reports, and more. Refer to the subsequent sections for step-by-step instructions on using the system to streamline inventory management within the ICT department.

Thank you for choosing the ICT Department Inventory System to enhance the efficiency and organization of your school's ICT resources. We are confident that this system will greatly benefit the ICT department and contribute to the overall success of your educational institution.

1.9.2 Login

To log in to the ICT Department Inventory System, follow these steps:

1. Launch the application by clicking the "Launch App" button found at the top of Sheet1.
2. A login form will appear, asking for user credentials.



UPDATES AVAILABLE Updates for Office are ready to be installed, but first we need to close some apps.

Username: emmanuel

Password: password

Name	Price	Qty	Weight(Kg	tion	Notes	Qty Available
jAC	100	1	2		No note mowSNDNVSD	10
jAC	100	1	2		No note mowSNDNVSD	10
jAC	100	1	2		No note mowSNDNVSD	10
dg	dga	da	asg	y Dam	da	

Figure 7: login Form

3. Enter “admin” for the username and “password” for the Password field.
4. Click the “Login” button at the bottom of the pop-up window.
5. If the information is entered correctly, you will be taken to the main view/dashboard.
6. If the information entered is incorrect, you will see a red text indicating that the information you entered is invalid. Please go back and refill the username and password fields correctly, and then retry step 4.

Please note that the provided username and password are for demonstration purposes only. In a live system, you would use the appropriate credentials provided by your system administrator.

1.9.3 Add Items to Inventory

To add items to the inventory, follow these steps:

1. Click the “Add” button at the top right of the main view/dashboard.
2. A new window will open with input fields to fill.

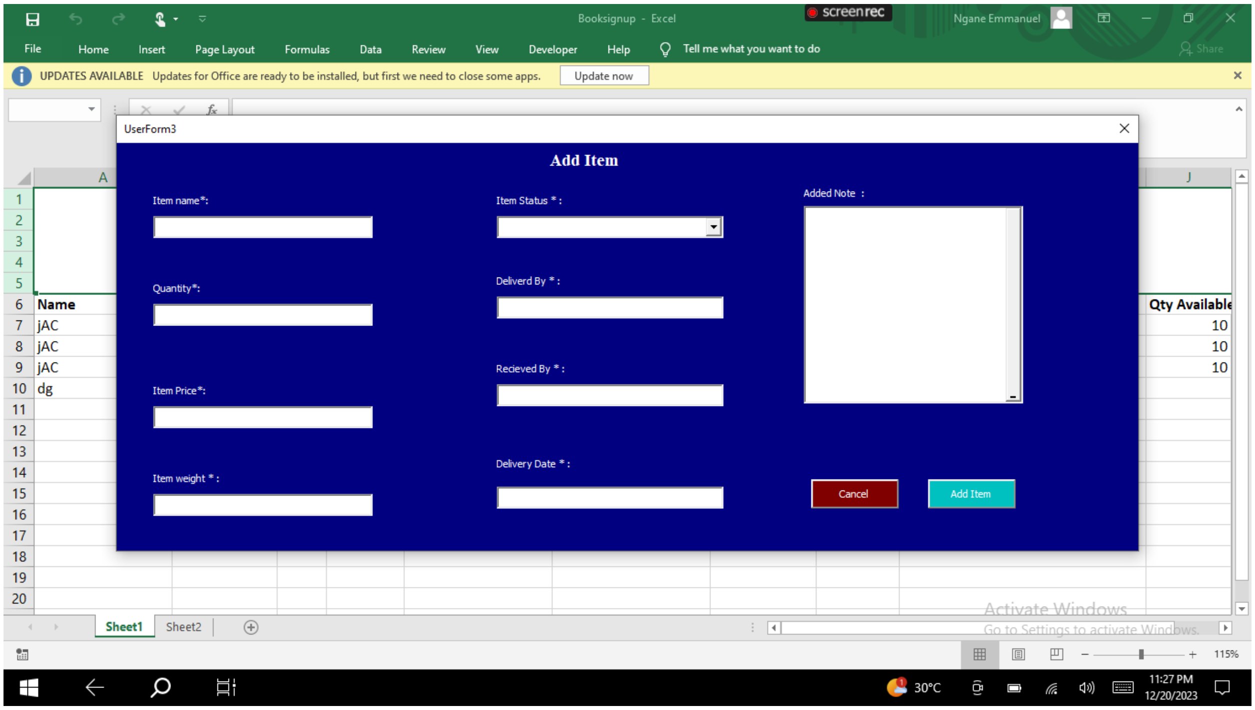


Figure 8: add items

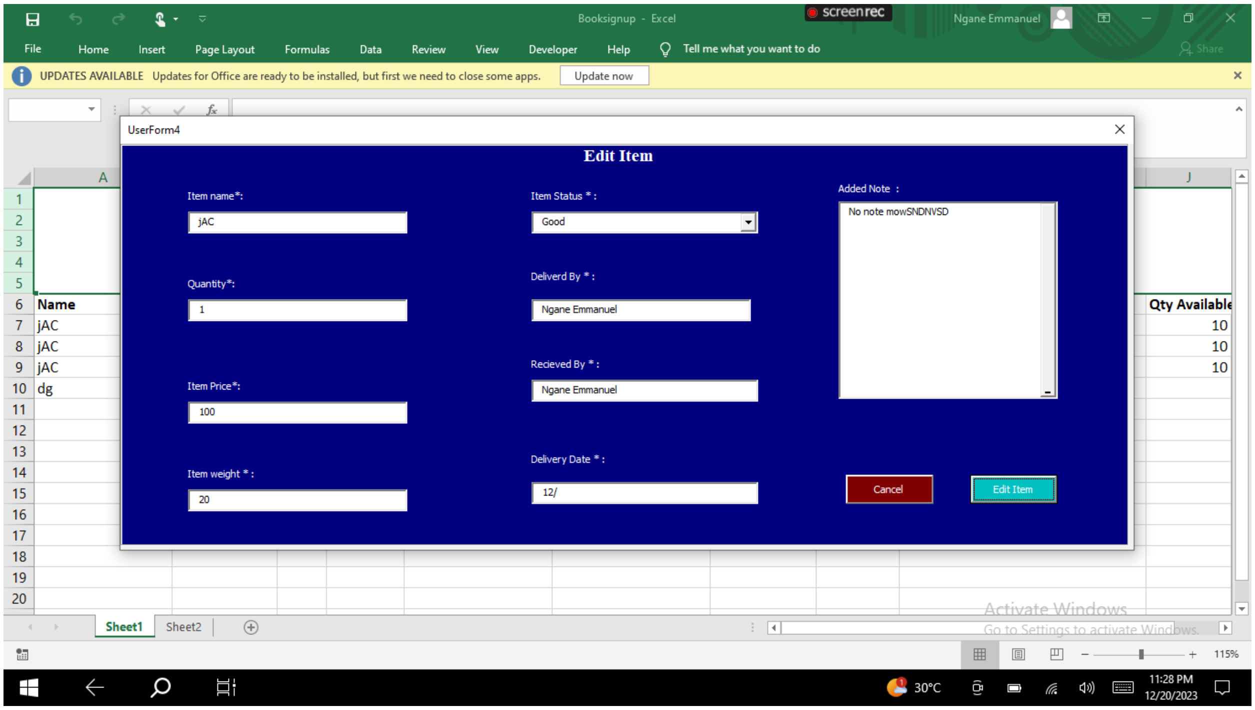
3. Fill in all the required fields denoted by (*).
4. Click the “Add” button at the bottom right.
5. You will receive a message indicating that the adding process was successful.
6. If you wish to cancel the add process, click the “Cancel” button located near the “Add” button.

Please ensure that you provide accurate and complete information while adding items to the inventory.

1.9.4 Edit Item in Inventory

To edit an existing item in the inventory, follow these steps:

1. Click on the “Inventory Item” tab in the main view.
2. Select the item you wish to edit by clicking on it.
3. Click the “Edit” button at the top right corner.
4. A window will open with pre-filled input boxes.



Edit Item

Item name*: jAC

Quantity*: 1

Item Price*: 100

Item weight * : 20

Item Status * : Good

Deliverd By * : Ngane Emmanuel

Recieved By * : Ngane Emmanuel

Delivery Date * : 12/

Added Note : No note mowSNDNVSD

	A	J
6	Name	Qty Available
7	jAC	10
8	jAC	10
9	jAC	10
10	dg	

Figure 9: Edit widnow

5. Change the required information in any input box.
6. Click the “Edit” button at the bottom right.
7. A message box should appear indicating that the item was successfully edited.
8. If you wish to cancel the editing process, click the “Cancel” button located near the “Edit” button.

Please ensure that you review and verify the changes before saving the edited item.

1.9.5 Delete Item in Inventory

To delete an item from the inventory, follow these steps:

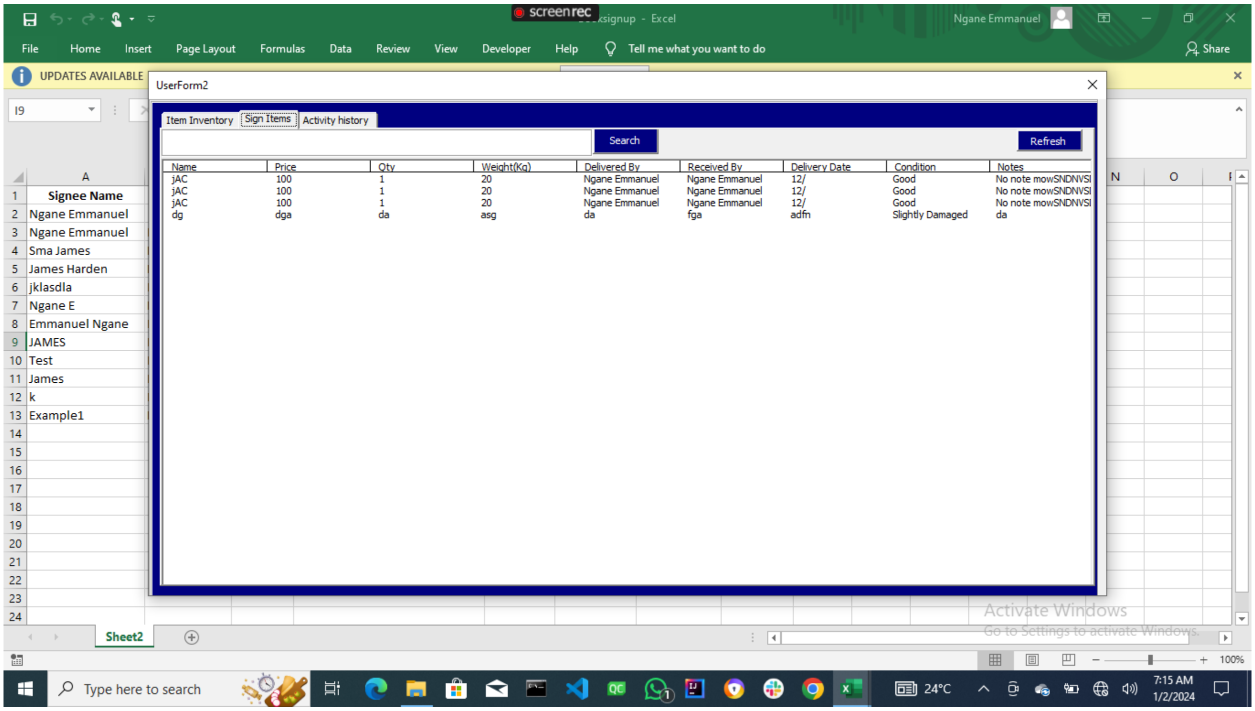
1. Click on the “Inventory Item” tab in the main view.
2. Select the item you wish to delete by clicking on it.
3. Click the “Delete” button at the top right corner.

Please exercise caution while deleting items as this action is irreversible.

1.9.6 Sign Out Items to Institution Members

To sign out items to institution members, follow these steps:

1. Click the “Sign Items” tab in the main view.



Name	Price	Qty	Weight(Kg)	Delivered By	Received By	Delivery Date	Condition	Notes
jAC	100	1	20	Ngane Emmanuel	Ngane Emmanuel	12/	Good	No note mowSNDNVSI
jAC	100	1	20	Ngane Emmanuel	Ngane Emmanuel	12/	Good	No note mowSNDNVSI
jAC	100	1	20	Ngane Emmanuel	Ngane Emmanuel	12/	Good	No note mowSNDNVSI
dg	dga	da	asg	da	fga	adfn	Slightly Damaged	da

	A
1	Signee Name
2	Ngane Emmanuel
3	Ngane Emmanuel
4	Sma James
5	James Harden
6	jklasdla
7	Ngane E
8	Emmanuel Ngane
9	JAMES
10	Test
11	James
12	k
13	Example1

Figure 10: Sign out items window

2. Select the item you wish to sign out by clicking on it.
3. Click the “Sign Out” button at the top right corner.
4. A new window will open with a list of institution members.
5. Select the member you wish to sign out the item to by clicking on their name.
6. Click the “Sign Out” button at the bottom right.
7. A message box should appear indicating that the item was successfully signed out.

Please ensure that you select the correct institution member and provide accurate information while signing out items.

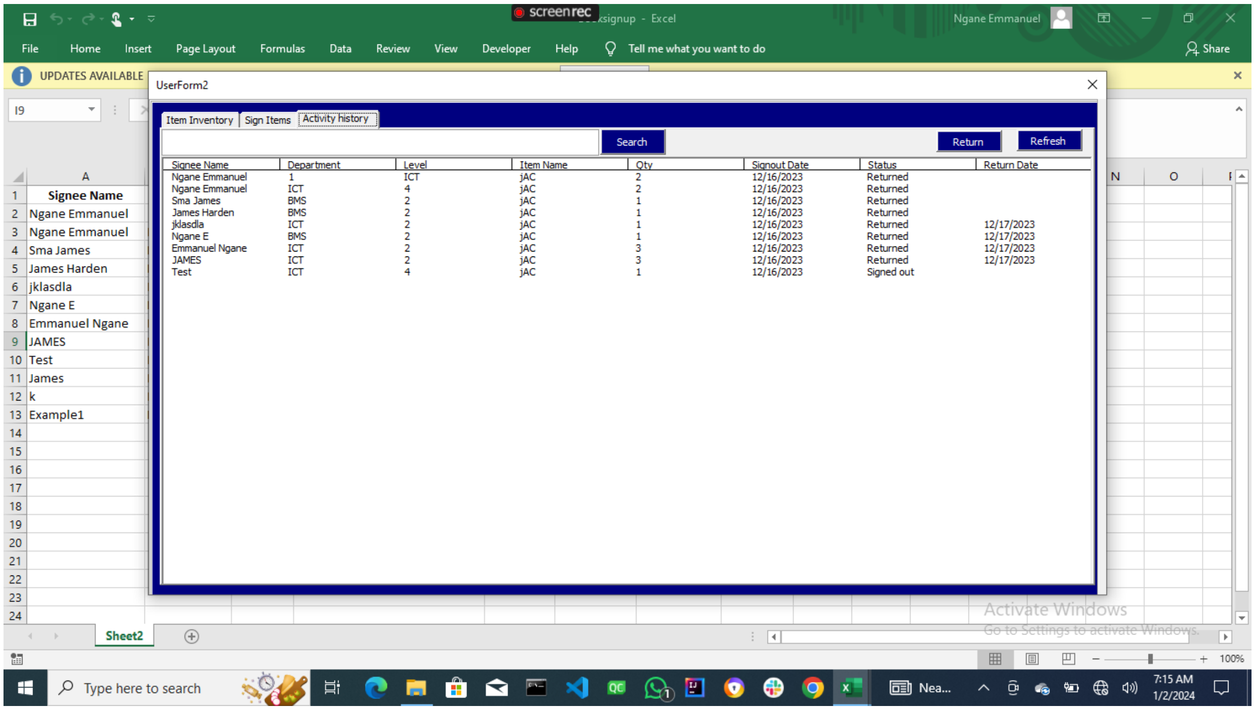
1.9.7 Return Items

2 8. Generate Reports

To return items back to the school storage, follow these steps:

1. Click the “Activities” tab in the main view.

2. Select the item you will like to return



Signee Name	Department	Level	Item Name	Qty	Signout Date	Status	Return Date
Ngane Emmanuel	1	ICT	jAC	2	12/16/2023	Returned	
Ngane Emmanuel	ICT	4	jAC	2	12/16/2023	Returned	
Sma James	BMS	2	jAC	1	12/16/2023	Returned	
James Harden	BMS	2	jAC	1	12/16/2023	Returned	
jklasdla	ICT	2	jAC	1	12/16/2023	Returned	12/17/2023
Ngane E	BMS	2	jAC	1	12/16/2023	Returned	12/17/2023
Emmanuel Ngane	ICT	2	jAC	3	12/16/2023	Returned	12/17/2023
JAMES	ICT	2	jAC	3	12/16/2023	Returned	12/17/2023
Test	ICT	4	jAC	1	12/16/2023	Signed out	

Figure 11: Activities view

3. Click the “Return” button
4. The status of the item should be changed to returned

Please note that the report generation process may take a few moments, depending on the size of the inventory data.

###. Troubleshooting If you encounter any issues or errors while using the ICT Department Inventory System, please follow these troubleshooting steps:

1. Check the error message displayed on the screen for any specific details about the problem.
2. Refer to the “FAQ” section of this user manual for common troubleshooting tips.
3. If the issue persists, contact your system administrator or IT support team for further assistance.

Please provide them with the necessary information, such as the steps leading to the error and any error messages displayed on the screen.

Thank you for using the ICT Department Inventory System. We hope this user manual has provided you with the necessary guidance to effectively manage and track your school’s ICT resources. If you have any questions or require further assistance, please don’t hesitate to reach out to our support team.

2.1 Conclusion

The ICT department inventory system provides an efficient and streamlined solution for managing inventory in educational institutions. Throughout the development and implementation process, several key aspects were addressed, including system design, software development, deployment, and user training. The successful implementation of the ICT department inventory system offers numerous benefits for the school, including improved inventory tracking, reduced manual effort, accurate reporting, and streamlined workflows.

By leveraging the system design phase, the requirements and functionalities of the ICT department inventory system were carefully analyzed and documented. This phase ensured a clear understanding of the system's scope and objectives, enabling the development team to design an effective and user-friendly solution.

During the software development phase, the system design was translated into working software. The development team followed coding guidelines and best practices to write high-quality, maintainable code. Thorough testing and debugging were performed to identify and rectify any issues or bugs, ensuring the robustness and reliability of the system.

The deployment phase involved installing the system on the target environment, configuring it to meet the school's specific requirements, and providing user training. Installation and configuration were executed following a systematic approach, ensuring a smooth setup process. User training sessions and materials were designed to familiarize end-users with the system's features and empower them to utilize it effectively.

User acceptance testing was a critical step in the deployment phase, where end-users performed testing to validate the system against their requirements and expectations. Any identified issues were promptly resolved, ensuring a high level of user satisfaction and system acceptance.

In conclusion, the ICT department inventory system has proven to be a valuable asset for educational institutions, facilitating efficient inventory management and enhancing overall productivity. The careful consideration given to system design, software development, deployment, and user training has resulted in a reliable and user-friendly solution that meets the unique needs of schools.

2.1.1 Summary

The ICT department inventory system is an essential tool for educational institutions to effectively manage their inventory. This system encompasses various functionalities, including inventory tracking, data validation, data storage, reporting, and user management. Throughout the development and implementation process, the ICT department inventory system underwent rigorous analysis, design, development, testing, deployment, and user training.

The system design phase focused on understanding the school's requirements and translating them into a comprehensive design specification. This phase ensured the system's functionality aligned with the school's inventory management needs.

Software development involved writing efficient and maintainable code to implement the system's functionalities. Thorough testing and debugging were conducted to ensure the system's reliability and to address any identified issues.

The deployment phase encompassed the installation and configuration of the system, as well as user training. The system was installed on the target environment, configured according to the school's specifications, and users were trained to effectively utilize the system's features.

User acceptance testing played a crucial role in validating the system's compliance with the school's requirements. Identified issues were resolved to ensure a satisfactory user experience.

Overall, the ICT department inventory system provides ICT University with a powerful tool to streamline their inventory management processes, reduce manual effort, and improve accuracy. The successful implementation of the system contributes to increased efficiency and productivity within the school's operations.