INVENTORY ORDER PLACING SYSTEM

A PROJECT REPORT

Submitted by

VINOTH RAJ G (210701509)

in partial fulfillment for the course

CS19P21 - ADVANCED ROBOTIC PROCESS AUTOMATION

for the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR

THANDALAM

CHENNAI – 602 105

NOVEMBER 2024

RAJALAKSHMI ENGINEERING COLLEGE

CHENNAI - 602105

BONAFIDE CERTIFICATE

Certified that this project report **"INVENTORY ORDER PLACING SYSTEM"** is the bonafide work of **"VINOTH RAJ G (210701509)"** who carried out the project work for the subject CS19P21- Advanced Robotic Process Automation under my supervision.

Dr. P.Kumar

HEAD OF THE DEPARTMENT

Professor and Head

Department of

Computer Science and Engineering

Rajalakshmi Engineering College

Rajalakshmi Nagar

Thandalam

Chennai - 602105

Mr. B.Bhuvaneswaran,

SUPERVISOR

Assistant Professor (SG),

Department of

Computer Science and Engineering

Rajalakshmi Engineering College

Rajalakshmi Nagar

Thandalam

Chennai - 602105

Submitted to Project and Viva Voce Examination for the subject CS19P21- Advanced Robotic Process Automation held on_____.

Internal Examiner **External Examiner**

ABSTRACT

The Inventory Order Placing System (IOPS) is a comprehensive software solution designed to streamline and optimize the process of managing inventory and placing orders within an organization. This system aims to enhance efficiency, accuracy, and overall effectiveness in inventory management by leveraging advanced technological features. IOPS provides a user-friendly interface accessible to authorized personnel, allowing them to view real-time inventory levels, track stock movements, and analyse historical data for informed decision-making. Through this system, users can automate the process of placing orders based on predefined thresholds or customized triggers, ensuring timely replenishment of stock while minimizing overstocking or stockouts.

Key functionalities of the IOPS include inventory monitoring, order generation, and vendor management. It enables users to categorize inventory items, set reorder points, and receive alerts when inventory levels fall below specified thresholds. Additionally, the system facilitates seamless communication with suppliers or vendors, automates order creation, and monitors order statuses from placement to delivery. IOPS is designed to integrate with existing enterprise resource planning (ERP) systems, enhancing interoperability and data synchronization across various departments or modules within the organization. Its scalability allows for customization and expansion to accommodate the evolving needs of different industries and business sizes.

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this report. Our sincere thanks to our Chairman **Mr. S.Meganathan, B.E, F.I.E.,** our Vice Chairman **Mr. Abhay Shankar Meganathan, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) Thangam Meganathan, Ph.D.,** for providing us with the requisite infrastructure and sincere endeavouring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N.Murugesan, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P.Kumar, M.E., Ph.D.**, Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guides, **Mr. B.Bhuvaneswaran, M.E.**, Assistant Professor (SG), and **Ms. J.Jinu Sophia, M.E.**, Assistant Professor (SG), Department of Computer Science and Engineering, Rajalakshmi Engineering College for their valuable guidance throughout the course of the project. We are very glad to thank our Project Coordinator, **Mr. B.Bhuvaneswaran, M.E.**, Assistant Professor (SG), Department of Computer Science and Engineering for his useful tips during our review to build our project.

VINOTH RAJ G (210701509)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION	ACCRONYM
RPA	Robotic Process Automation
URL	Uniform Resource Locator

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Inventory is the stock of goods and materials that a business or organization holds for the purpose of production or sale. Inventory management is the process of planning, organizing, and controlling the flow of inventory from purchase to sale. Inventory management is essential for reducing costs, increasing efficiency, and enhancing customer satisfaction. However, inventory management also involves various challenges and risks, such as inventory fluctuations, uncertainties in demand and supply, wastage and losses, etc.

Therefore, effective inventory management requires applying various techniques and tools to optimize inventory levels and minimize inventory-related problems. One of the important techniques and tools for inventory management is inventory order placing, which is the process of placing orders for inventory items based on the current stock level and the demand forecast.

Inventory order placing can help to ensure a continuous supply of materials and stock so that production and customer demand are not affected by shortages or delays. It can also help to avoid both overstocking and understocking of inventory, which can result in higher costs, lower profits, and lower customer satisfaction. However, inventory order placing is also a complex and time-consuming task, which requires a lot of data, analysis, and decision making.

Therefore, inventory order placing can benefit from automation, which is the use of technology and software to perform tasks without human intervention. An inventory order placing system is a system that automates the process of placing orders for inventory items based on the current stock level and the demand forecast. An inventory order placing system can have many benefits for both customers and businesses, such as reducing human errors and delays,

improving inventory accuracy and efficiency, saving time and money, and enhancing customer satisfaction and loyalty. This paper aims to design and implement an inventory order placing system using UiPath, which is a software platform that enables robotic process automation (RPA), which is the use of software robots to perform repetitive and rule-based tasks.

The paper will cover the following aspects of the inventory order placing system: data collection, data processing, data analysis, data output, and data validation. The paper will also evaluate the performance and accuracy of the inventory order placing system and discuss the limitations and future work of the system.

UiPath is a software company that makes robotic process automation (RPA) software. RPA is a technology that uses software robots to automate repetitive and manual tasks that humans usually do. Some of the features of UiPath are:

- **UiPath Studio:** This is the tool that allows users to design and develop automation workflows using a graphical user interface. Users can drag and drop activities, variables, and data types to create sequences, flowcharts, state machines, or global exception handlers. Users can also use programming languages such as C# or VB.NET to write custom code. UiPath Studio also provides debugging and testing features to ensure the quality and reliability of the workflows.

- **UiPath Robot:** This is the tool that executes the automation workflows created in UiPath Studio. Users can run the robots in two modes: attended or unattended. Attended robots run on the user's machine and require human intervention or supervision. Unattended robots run on a server or a virtual machine and can operate without human involvement. Users can also schedule, monitor, and manage the robots using UiPath Orchestrator.

- **UiPath Orchestrator:** This is the tool that provides a centralized platform to manage, monitor, and optimize the automation processes. Users can deploy, configure, and control the robots across multiple environments using a web-based dashboard. Users can also track the performance, logs, and audit trails of the robots and generate reports and analytics. UiPath Orchestrator also enables users to integrate the automation workflows with other systems and applications using APIs, webhooks, or queues.

- **UiPath Apps:** This is the tool that allows users to create and deploy user interfaces for the automation workflows. Users can design and build web or mobile apps using a drag and drop interface and connect them to the robots using UiPath Data Service. Users can also interact with the robots using voice, chat, or gestures and provide feedback or inputs. UiPath Apps also supports collaboration and sharing features to enable users to work together on the app development.

These are some of the main features of UiPath that make it a powerful and versatile RPA software. UiPath aims to help organizations achieve digital transformation and improve their efficiency and productivity.

1.2 OBJECTIVE

- **Efficiency Enhancement:** Streamline the ordering process to minimize manual effort and time spent on placing orders.
- **Inventory Optimization:** Ensure optimal inventory levels by facilitating accurate and timely ordering of goods and materials.
- **Real-Time Monitoring:** Enable real-time monitoring of stock levels to prevent overstocking or stockouts.
- **Vendor Management:** Facilitate seamless communication and interaction with vendors for order placement, tracking, and confirmation.
- **Data Accuracy:** Ensure data accuracy by implementing robust validation

mechanisms for orders, reducing errors in ordering and inventory management.

- **Integration Capabilities:** Enable integration with existing inventory management systems for seamless data flow and synchronization.
- **User-Friendly Interface:** Develop an intuitive interface for easy navigation and order placement, reducing training time for users.
- **Reporting and Analytics:** Provide comprehensive reporting and analytical tools to track order history, supplier performance, and inventory turnover.
- **Cost Optimization:** Optimize costs by automating routine tasks, reducing human errors, and negotiating better terms with vendors based on historical data.
- **Scalability and Flexibility:** Design a system capable of scaling with business growth and adaptable to changing inventory requirements and processes.

These objectives aim to create an efficient, reliable, and adaptable inventory order placing system that benefits the organization in managing its inventory effectively.

1.3 EXISTING SYSTEM

The existing system for an inventory order placing system might vary depending on the organization's setup, but typically it involves a combination of manual and semi-automated processes:

- **Manual Ordering:** Currently, orders might be placed manually through phone calls, emails, or physical order forms filled out by employees. This method is time-consuming and prone to errors.
- **Spreadsheets or Basic Software:** Some businesses might use

spreadsheets or basic software tools to track inventory levels and manually calculate when to reorder items. These systems often lack real-time updates and integration capabilities.

- **Limited Integration:** Existing systems might have limited integration between inventory management and order placing. Data might be siloed, leading to inefficiencies and inaccuracies in order fulfilment.
- **Vendor Communication:** Communication with vendors might be fragmented, leading to delays or misunderstandings in the ordering process. Confirmations and updates might be done manually and lack real-time visibility.
- **Lack of Analytics:** There might be a lack of robust analytics and reporting tools to analyse historical data, supplier performance, and inventory trends, hindering informed decision-making.
- **Potential for Errors:** Manual data entry increases the chances of errors in orders, leading to overstocking, stockouts, or incorrect deliveries, impacting operational efficiency and customer satisfaction.
- **Limited Scalability:** The current system may struggle to scale with the growing needs of the business. As order volumes increase, the manual process becomes even more burdensome and error-prone.

Improving upon these existing system elements can greatly enhance the efficiency, accuracy, and agility of the inventory order placing process within an organization. Integration of technology, automation, and better communication channels with vendors are some areas that could significantly enhance the system.

1.4 PFROPOSED SYSTEM

The proposed system for an inventory order placing system using UiPath, a popular Robotic Process Automation (RPA) tool, involves the following components and functionalities:

- **UiPath Orchestrator:** Utilize the Orchestrator as the centralized platform to manage, schedule, and monitor the bots performing inventory order placements.
- **Bot Creation and Configuration:** Develop UiPath bots tailored for different aspects of the order placing process, such as order initiation, vendor communication, order tracking, and confirmation.
- **Integration with ERP/Inventory Systems:** Integrate UiPath bots with existing ERP or inventory management systems to access real-time inventory levels, supplier information, and order history.
- **Automated Order Generation:** Create bots to automatically generate purchase orders based on predefined triggers like inventory thresholds, historical data, or demand forecasts.
- **Vendor Communication and Order Submission:** Use bots to communicate with vendors via emails, portals, or APIs to place orders, ensuring accuracy and timeliness in the submission process.
- **Error Handling and Validation:** Implement error handling mechanisms within bots to address exceptions or discrepancies during the order placement process, ensuring data accuracy and completeness.
- **Order Tracking and Confirmation:** Develop bots to track orders in transit and confirm delivery against the purchase orders, updating the inventory systems in real-time.
- **Reporting and Analytics:** Configure UiPath bots to collect order data and generate reports providing insights into order volumes, lead times, vendor

performance, and inventory turnover.

- **Security and Compliance:** Implement security measures to ensure data privacy and compliance with relevant regulations, especially concerning sensitive information shared during the order process.
- **Testing and Maintenance:** Regularly test and maintain the bots to ensure they function efficiently, incorporating updates or modifications based on changing business requirements.

By leveraging UiPath's capabilities, the proposed inventory order placing system aims to automate and streamline the order management process, reducing manual efforts, minimizing errors, and enhancing overall efficiency in inventory operations

CHAPTER 2

LITERATURE REVIEW

Inventory order placing system is a system that automates the process of placing orders for inventory items based on the current stock level and the demand forecast. Inventory order placing system is essential for reducing costs, increasing efficiency, and enhancing customer satisfaction. However, inventory order placing system also involves various challenges and risks, such as inventory fluctuations, uncertainties in demand and supply, wastage and losses, etc. Therefore, effective inventory order placing system requires applying various techniques and tools to optimize inventory levels and minimize inventory-related problems. This report aims to review and analyse the existing research and literature on the topic of inventory order placing system.

The study by "**Hansen, O., Transchel, S., & Friedrich, H. (2023). Replenishment strategies for lost sales inventory systems of perishables under demand and lead time uncertainty. European Journal of Operational Research, 308(2), 661-675**", delves into optimizing replenishment strategies for perishable goods in lost sales inventory systems amid uncertainties in both demand and lead times. The researchers focus on addressing the challenges faced by industries dealing with perishable items where inventory cannot be stored indefinitely due to expiration. They aim to devise strategies that ensure minimized losses while meeting demand fluctuations and accounting for unpredictable lead times.

The paper starts by highlighting the complexity of managing perishable inventory systems, where demand is uncertain, and lead times for replenishment can vary. It emphasizes the importance of incorporating these uncertainties into decision-making processes to avoid stockouts or excessive wastage. The study

then proposes a mathematical model that integrates demand and lead time uncertainties into the optimization of replenishment decisions. This model aims to strike a balance between the cost of holding inventory and the potential revenue loss from stockouts.

Through extensive simulations and analyses, the researchers demonstrate the effectiveness of their proposed model in managing perishable inventory systems. They compare various replenishment strategies, considering different levels of uncertainty, and evaluate their performance in terms of minimizing overall costs and mitigating the risk of stockouts. The findings highlight the significance of adaptive strategies that dynamically adjust inventory levels based on real-time demand and lead time information, showcasing their superiority in handling uncertainties over static approaches.

The study by "**Kumar, A., Santra, P. K., & Mahapatra, G. S. (2023). Fractional order inventory system for time-dependent demand influenced by reliability and memory effect of promotional efforts. Computers & Industrial Engineering, 179, 109191**", their research delves into the complexities of managing inventory under time-dependent demand scenarios, considering the impact of reliability and the memory effect of promotional efforts. The fractional-order aspect introduces a new dimension to traditional inventory models, emphasizing the non-integer nature of inventory dynamics.

The paper investigates the influence of time-varying demand patterns, emphasizing the need for a system that can adapt to these fluctuations effectively. By integrating the reliability factor and the memory effect of promotional activities, the study aims to enhance the accuracy of inventory control models. This approach aligns with the dynamic nature of markets, where demand patterns are often influenced by promotional campaigns and changing consumer behavior.

The research methodology involves the formulation of a fractional-order differential equation to model the inventory system. Through simulations and analysis, the authors demonstrate the efficacy of this approach in optimizing inventory decisions under time-dependent demand, emphasizing the significance of considering reliability and the memory effect of promotions. Their findings showcase the potential of fractional-order models in addressing the complexities inherent in contemporary inventory management systems.

The research article by "**Berling, P., Johansson, L., & Marklund, J. (2023). Controlling inventories in omni/multi-channel distribution systems with variable customer order-sizes. Omega, 114, 102745**", specifically addressing the challenge of varying customer order sizes. The authors recognize the complexity of modern distribution systems, where customers expect seamless experiences regardless of the channel, they use to place orders. One key issue they tackle is the variability in order sizes across different channels, which significantly impacts inventory control.

The study focuses on devising effective inventory management models that can adapt to these fluctuating order sizes. By acknowledging the diversity in customer behaviours across channels, the researchers propose a model that considers this variability. They introduce a novel approach that integrates demand estimation techniques and optimization algorithms to optimize inventory levels across multiple channels. This model aims to strike a balance between fulfilling customer orders promptly and minimizing inventory holding costs.

Through extensive simulations and analysis, Berling, Johansson, and Marklund demonstrate the efficacy of their proposed approach. Their results highlight the potential for significant improvements in inventory management within omni/multi-channel distribution systems. By adopting their model, businesses can better anticipate demand variations, optimize inventory levels.

CHAPTER 3

SYSTEM DESIGN

3.1 SYSTEM FLOW DIAGRAM

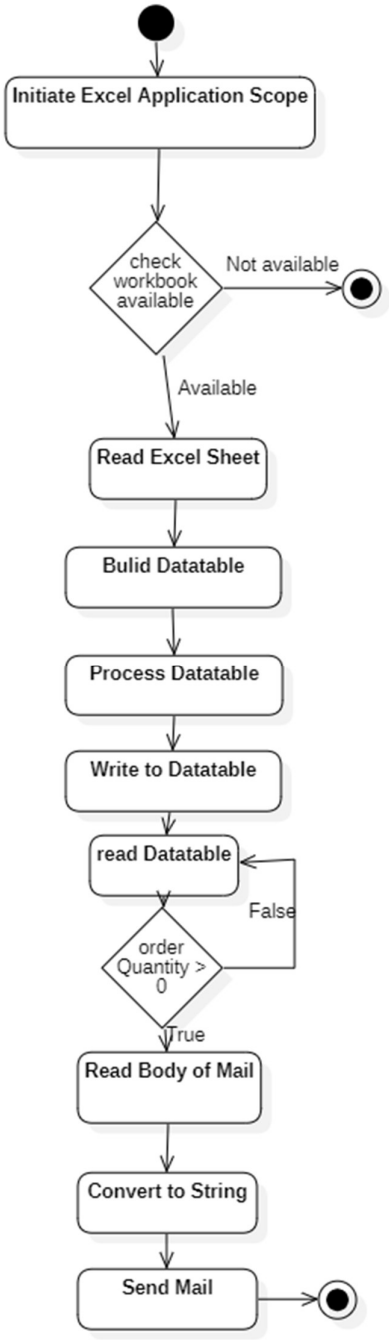


Fig 3.1 System Flow Diagram

A flowchart is a type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. The system flow diagram illustrates the streamlined process of the Inventory Order Placing System powered by UiPath. The flow begins as the user initiates an order request through the inventory system. The UiPath bots come into action, triggered by this request, accessing inventory data within the system. They communicate seamlessly with vendor systems, sending order details and receiving confirmations. Concurrently, the inventory system updates the order status and informs the user about successful order placement. Throughout the process, the UiPath bots play a pivotal role in orchestrating communication between the user, inventory system, and vendor system, ensuring real-time updates, accurate order placement, and efficient order tracking. This iterative process encompasses data validation, order generation, vendor communication, and status updates, all orchestrated through UiPath's automation capabilities. The system flow diagram captures this cohesive interaction, showcasing the integration of UiPath bots within the inventory order placement process, ultimately optimizing efficiency and accuracy while reducing manual intervention is described in the above figure 3.1.

3.2 ARCHITECTURE DIAGARM

The architecture diagram for the Inventory Order Placing System leveraging UiPath encompasses a cohesive framework designed to automate and optimize the entire order placement process. At its core lies the UiPath RPA platform, serving as the orchestrator of automated tasks and interactions across various components.

The system architecture comprises multiple layers interconnected for seamless functionality. The front-end interface allows users, including inventory managers and procurement teams, to initiate and oversee order placements. This

interface interacts with the UiPath Orchestrator, serving as the central hub for managing, scheduling, and monitoring UiPath bots.

The UiPath bots, the backbone of automation, facilitate communication with the inventory system for real-time data access regarding stock levels, item details, and order history. These bots orchestrate the order generation, vendor communication, and order status updates, ensuring accuracy and timeliness in the entire procurement process.

Furthermore, the architecture integrates with ERP and vendor systems for data exchange, leveraging APIs or integration modules. The system utilizes a database to store and manage order-related information, enabling reporting, analytics, and future enhancements is described in below Figure 3.2

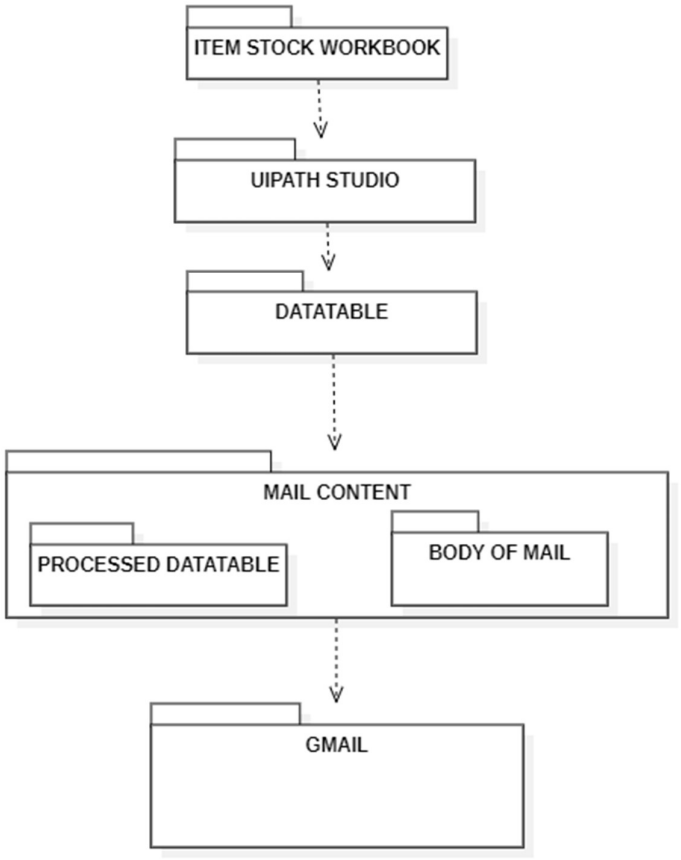


Fig 3.2 Architecture Diagram

3.3 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. The sequence diagram for the Inventory Order Placing System using UiPath encapsulates the orchestrated interactions between various components in the order placement process. It commences with the user initiating an order request within the Inventory System, triggering the UiPath Bot. The bot accesses inventory data, communicates with the Vendor System to send order details, and awaits confirmation. Upon reception, the Vendor System confirms the order, prompting the UiPath Bot to update the Inventory System and notify the user of the order placement status. The Inventory System, in turn, communicates the confirmation back to the user. This sequential flow of interactions depicts a streamlined process, initiating from user action, progressing through automated steps facilitated by UiPath, and culminating in the update and confirmation loop between systems and the user. This diagram highlights the orchestration of events, showcasing the seamless communication and data flow orchestrated by UiPath, pivotal in automating and optimizing the inventory order placement process for efficient and accurate procurement is described in the below Figure 3.3.

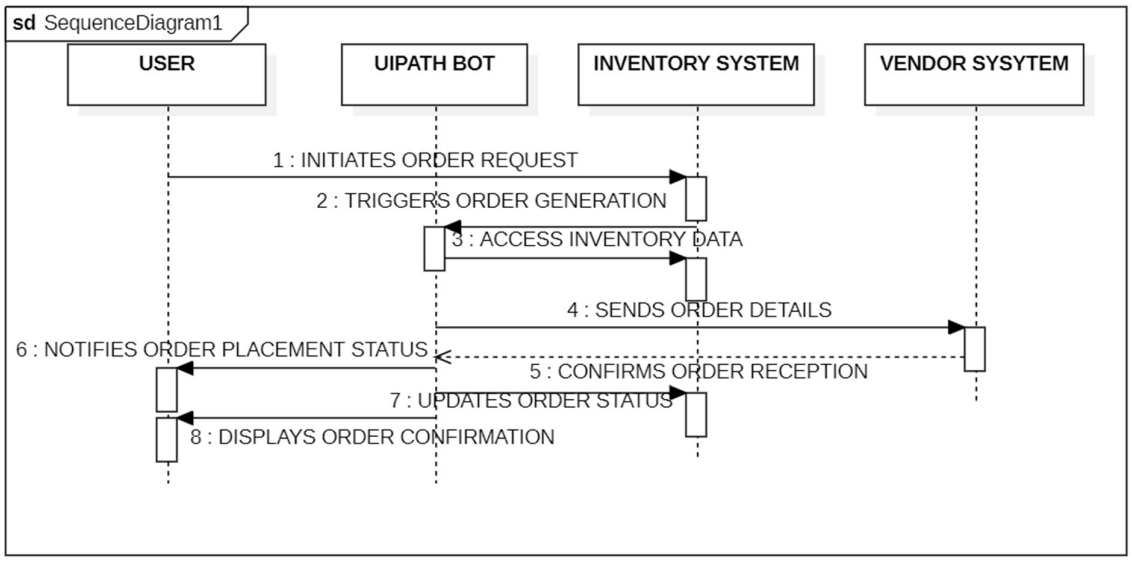


Fig 3.3 Sequence Diagram

CHAPTER 4

PROJECT DESCRIPTION

4.1 MODULES

4.1.1 SYSTEM USER

The user of an Inventory Order Placing System powered by UiPath could be:

- **Inventory Managers:** They use the system to oversee stock levels, initiate and monitor order placements, ensuring optimal inventory levels are maintained.
- **Procurement Team:** Responsible for managing vendor relationships, they utilize the system to streamline communication and ordering processes, ensuring timely and accurate procurement.
- **Warehouse Personnel:** Use the system to anticipate incoming inventory, manage space allocation, and prepare for incoming stock, improving warehouse efficiency.
- **Finance Department:** Utilizes the system to track and manage purchase orders, ensuring alignment with budgetary constraints and optimizing spending.
- **IT Administrators:** Responsible for system maintenance, they oversee the functionality, updates, and integration of the UiPath-powered system with existing inventory and ERP systems.
- **Analytics Teams:** Utilize the system's data to generate reports, analyze trends, and make informed decisions regarding inventory management strategies.
- **Operations Managers:** Rely on the system to ensure smooth order processing, minimizing delays, and optimizing operational workflows.
- **Quality Control:** Utilizes the system to monitor incoming inventory, ensuring compliance with quality standards and regulations.
- **Logistics Teams:** Utilize the system to coordinate with vendors, track

shipments, and manage the logistics of incoming inventory.

- **Executive Leadership:** Relies on the system's insights and reports to make strategic decisions regarding inventory investment, supplier relationships, and operational efficiencies.

In essence, various stakeholders across different departments and functions within an organization leverage the Inventory Order Placing System powered by UiPath to streamline, optimize, and manage the entire inventory procurement process efficiently.

4.1.2 EXCEL APPLICATION SCOPE

The Excel Application Scope activity in UiPath is used to automate interactions with Excel files. Here's a simplified explanation of its use within the context of an Inventory Order Placing System:

- **Purpose:** The Excel Application Scope is employed to perform actions like reading data from Excel files, writing data into Excel sheets, or manipulating Excel data during the order placement process.
- **Configuration:** Within the activity, specify the path to the Excel file that contains relevant inventory or order data. This could include details like item codes, quantities, vendors, etc.
- **Read or Write Actions:** Use activities like Read Range to extract data from Excel sheets into variables, or Write Range to update Excel files with new data (e.g., updated inventory levels, order information).
- **Data Manipulation:** Perform data manipulations such as filtering, sorting, or formatting using UiPath activities within the Excel Application Scope.
- **Error Handling:** Implement error handling mechanisms within the scope to manage exceptions that might occur during Excel file interactions.

4.1.3 GMAIL APPLICATION SCOPE

In an inventory order placing system using UiPath, the Gmail Application Scope can simplify email-related tasks such as communication with vendors or

order confirmation. Here's a simplified breakdown of how to use Gmail Application Scope:

1. **Scope Initialization:** Use the "Gmail Application Scope" activity to initialize the connection to Gmail. Drag and drop this activity into the UiPath sequence or workflow.
2. **Authentication:** Within the activity properties, input the Gmail account credentials (username and password) to authenticate the UiPath robot to access the Gmail account.
3. **Email Operations:**
    - **Reading Emails:** Use the "Get IMAP Mail Messages" activity within the Gmail Application Scope to retrieve emails from specific folders based on predefined criteria like subject lines, sender, or keywords.
    - **Sending Emails:** Utilize the "Send SMTP Mail Message" activity within the scope to compose and send emails to vendors or relevant stakeholders. Set recipient addresses, subject, body, and attachments as needed.
4. **Handling Emails:**
    - **Processing Emails:** Implement logic to parse and process the retrieved emails. For instance, extracting order confirmations or relevant information.
    - **Responding to Emails:** Set up actions or conditions to trigger responses based on received emails, such as confirming orders, acknowledging receipt, or flagging for further action.
5. **Error Handling:**
    - Implement error handling mechanisms within the scope to manage exceptions during email operations, ensuring the workflow continues smoothly even in case of email-related issues.

6. **Scope Termination:**
    - Finally, ensure to properly close the Gmail Application Scope activity to release resources and disconnect from the Gmail account.

By utilizing the Gmail Application Scope in UiPath, you can easily integrate email functionalities into the workflow of your inventory order placing system, facilitating efficient communication with vendors and handling email-based order confirmations or inquiries.

4.1.4 DATATABLE

In an inventory order placing system, using a DataTable within UiPath can simplify data handling, manipulation, and storage. Here is a simplified example of how a DataTable could be utilized within the project:

1. **Storing Order Information:**
    - Use a DataTable to store order details such as order ID, item name, quantity, vendor information, order status, etc.
    - Add rows to the DataTable to populate it with order information obtained from user input or external systems.
2. **Data Manipulation and Filtering:**
    - Perform operations like sorting, filtering, and searching within the DataTable to find specific orders, track inventory levels, or identify pending orders.
3. **Data Validation:**
    - Validate the entered data against the existing DataTable to ensure accuracy and prevent duplication or inconsistencies in orders.
4. **Integration with ERP/Inventory Systems:**
    - Use DataTables to structure and organize data fetched from ERP or inventory systems for further processing or comparison with existing orders.
5. **Reporting and Analytics:**
    - Generate reports by leveraging DataTable functionalities to

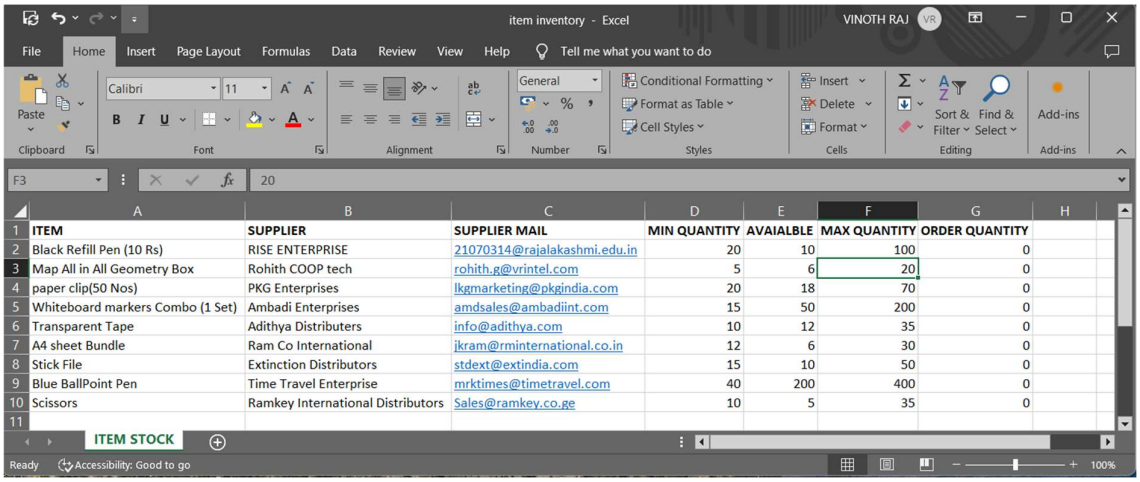
aggregate, summarize, or analyse order data. For instance, calculate total order quantities, identify top-selling items, or track vendor performance.

6. **Order Status Updates:**
    - Update the DataTable to reflect changes in order status, such as order placement, delivery confirmation, or cancellation.
7. **Iterating Through Orders:**
    - Use loops to iterate through the DataTable rows, performing actions on each order entry, such as sending emails to vendors, updating inventory systems, or notifying users about order status changes.

CHAPTER 5

OUTPUT SCREENSHOT

The data for the processing for the inventory order placing system is collected through the excel sheet mentioned below in Figure 5.1.



	A	B	C	D	E	F	G	H
1	ITEM	SUPPLIER	SUPPLIER MAIL	MIN QUANTITY	AVAIALBLE	MAX QUANTITY	ORDER QUANTITY	
2	Black Refill Pen (10 Rs)	RISE ENTERPRISE	21070314@rajalakashmi.edu.in	20	10	100	0	
3	Map All in All Geometry Box	Rohith COOP tech	rohith.g@vrintel.com	5	6	20	0	
4	paper clip(50 Nos)	PKG Enterprises	lkgmarketing@pkgindia.com	20	18	70	0	
5	Whiteboard markers Combo (1 Set)	Ambadi Enterprises	amdsales@ambadiint.com	15	50	200	0	
6	Transparent Tape	Adithya Distributers	info@adithya.com	10	12	35	0	
7	A4 sheet Bundle	Ram Co International	jkram@rminternational.co.in	12	6	30	0	
8	Stick File	Extinction Distributors	stdext@extindia.com	15	10	50	0	
9	Blue BallPoint Pen	Time Travel Enterprise	mrktimes@timetravel.com	40	200	400	0	
10	Scissors	Ramkey International Distributors	Sales@ramkey.co.ge	10	5	35	0	
11								

Fig 5.1 Input Excel Sheet

The processed data from the excel which is converted into datatable during the processing stage is shown to the user who initiated the system as output through the message box in below Figure 5.2.

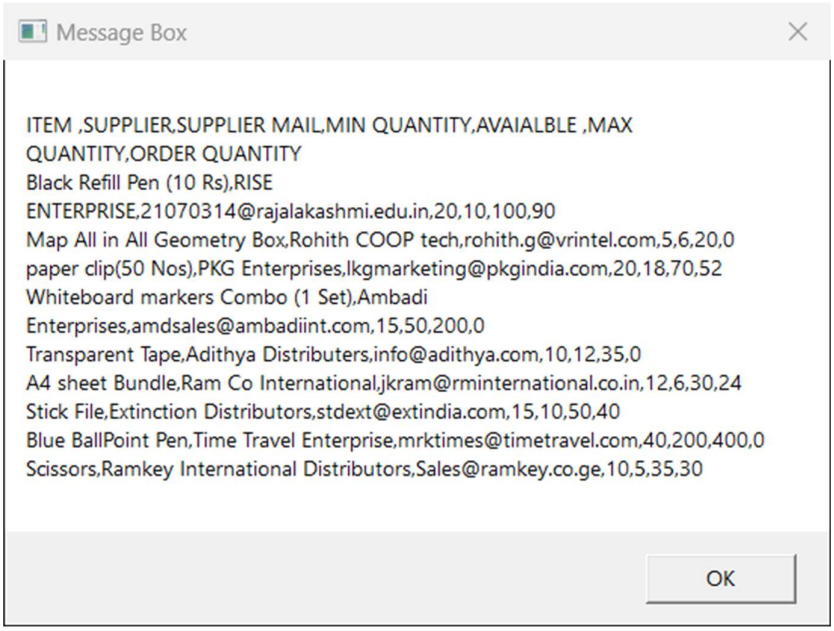


Fig 5.2 Processed DataTable Result in Message Box

Based on the processed datatale result, the gmail will be send to appropriate supplier based on the quantity derived in ordered quantity is shown below in Figure 5.3

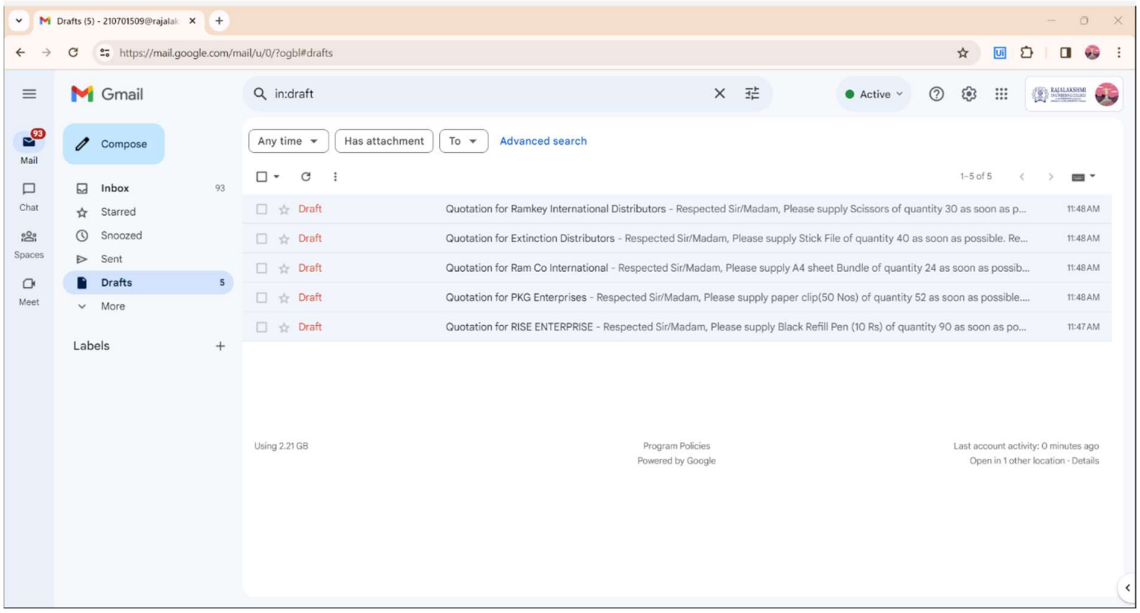


Fig 5.3 Mail Created and Saved in Draft

The mail is created and sent by content from the notepad as well as the processed datatable's content is shown in the below Figure 5.4.

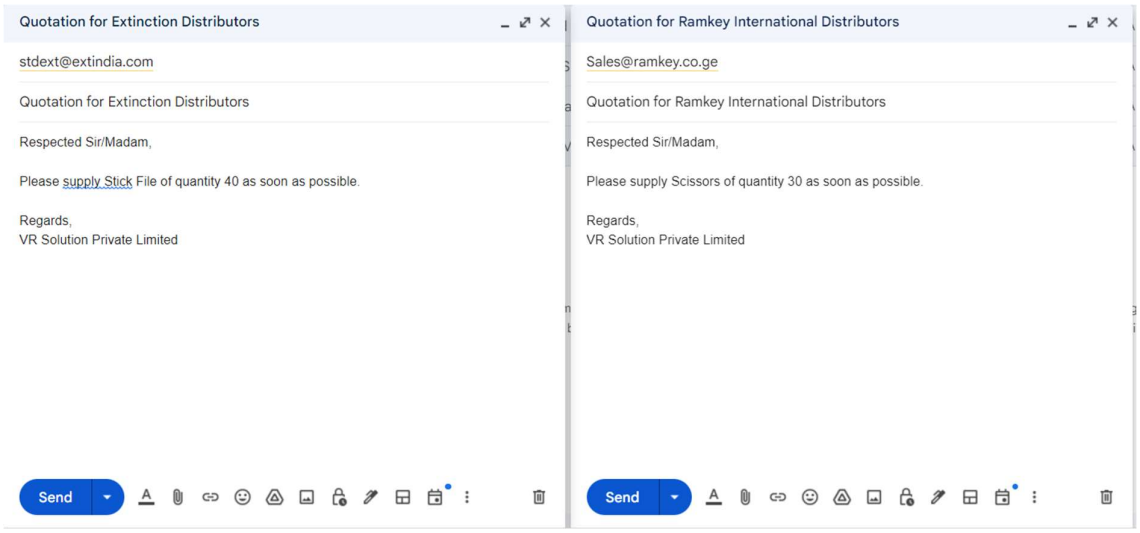
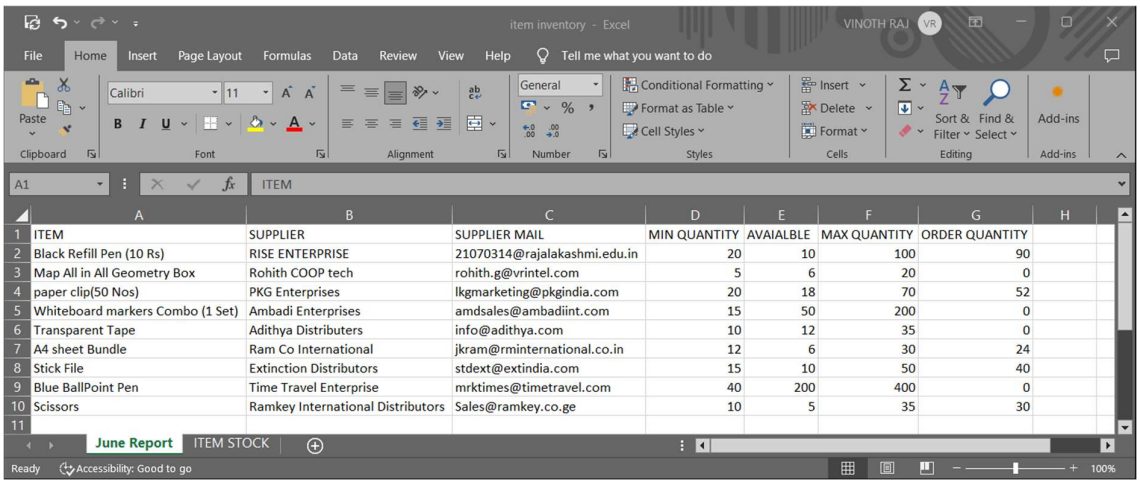


Fig 5.4 Mail Created with Content

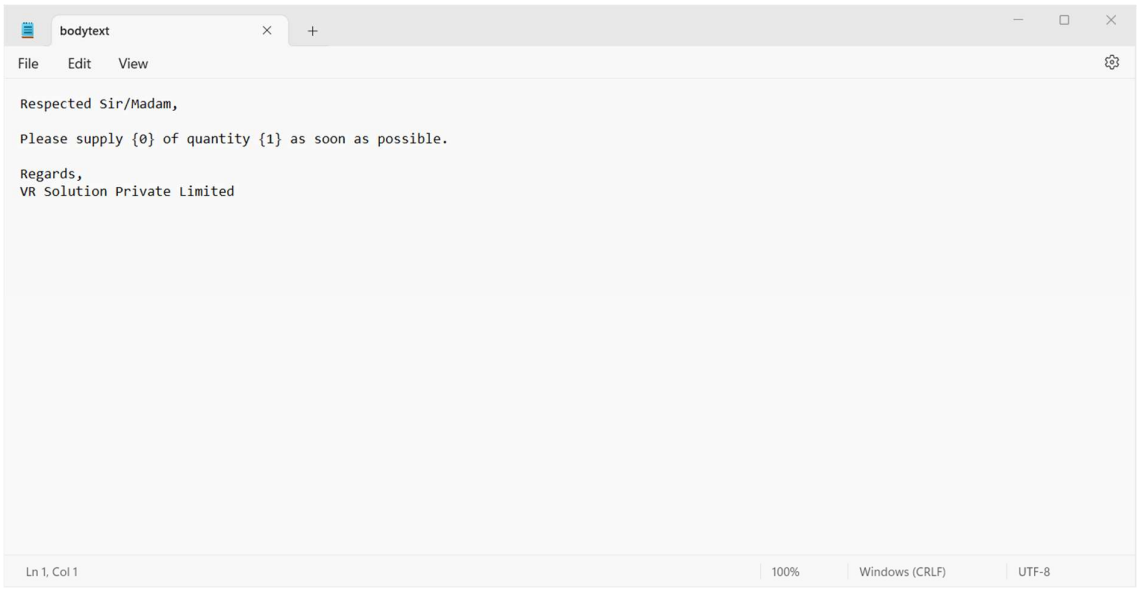
The processed datatable is stored in the new Excel sheet in the same workbook is shown in the below Figure 5.5.



	A	B	C	D	E	F	G	H
1	ITEM	SUPPLIER	SUPPLIER MAIL	MIN QUANTITY	AVAIALBLE	MAX QUANTITY	ORDER QUANTITY	
2	Black Refill Pen (10 Rs)	RISE ENTERPRISE	21070314@rajalakashmi.edu.in	20	10	100	90	
3	Map All in All Geometry Box	Rohith COOP tech	rohith.g@vrintel.com	5	6	20	0	
4	paper clip(50 Nos)	PKG Enterprises	lkgmarketing@pkgindia.com	20	18	70	52	
5	Whiteboard markers Combo (1 Set)	Ambadi Enterprises	amdsales@ambadiint.com	15	50	200	0	
6	Transparent Tape	Adithya Distributers	info@adithya.com	10	12	35	0	
7	A4 sheet Bundle	Ram Co International	jkram@rminternational.co.in	12	6	30	24	
8	Stick File	Extinction Distributors	stdext@extindia.com	15	10	50	40	
9	Blue BallPoint Pen	Time Travel Enterprise	mrktimes@timetravel.com	40	200	400	0	
10	Scissors	Ramkey International Distributors	Sales@ramkey.co.ge	10	5	35	30	
11								

Fig 5.5 Processed Data table stored in New Excel Sheet

The body of the mail content used for the gmail which will vary on each supplier is shown in the below Figure 5.6.



```
Respected Sir/Madam,

Please supply {0} of quantity {1} as soon as possible.

Regards,
VR Solution Private Limited
```

Fig 5.6 Body of the Mail in Text

The overall completion of the inventory order placing system is shown through the message box as completion message is shown in the below Figure 5.7.

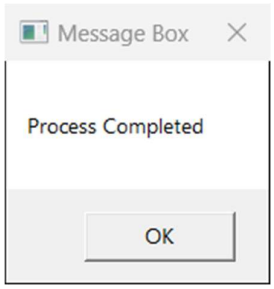


Fig 5.7 Process Completion

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

The implementation of the Inventory Order Placing System leveraging UiPath has revolutionized our inventory management processes. The automation facilitated by UiPath's robust functionalities has significantly enhanced operational efficiency, minimized errors, and streamlined the order placement workflow. Through this system, we have achieved:

1. **Enhanced Efficiency:** Automation of order generation, vendor communication, and real-time updates has expedited the order placement process, reduced manual efforts, and improved turnaround times.
2. **Accurate Inventory Management:** Real-time inventory tracking and data validation have led to better inventory control, minimizing stockouts and overstocking situations.
3. **Cost Optimization:** Reduced lead times, minimized errors, and better vendor management have contributed to cost savings and improved budgetary control.
4. **Improved Decision-Making:** Access to comprehensive reports and analytics has empowered informed decision-making, aiding in strategizing inventory procurement and vendor relationships.

6.2 FUTURE ENHANCEMENTS

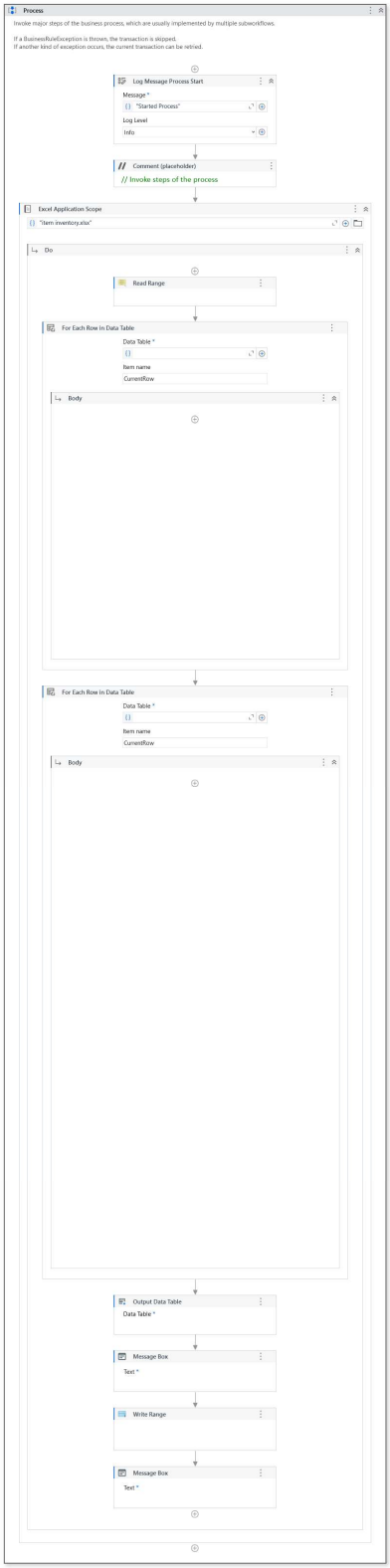
While the current system has delivered substantial benefits, there remain opportunities for further enhancements and advancements:

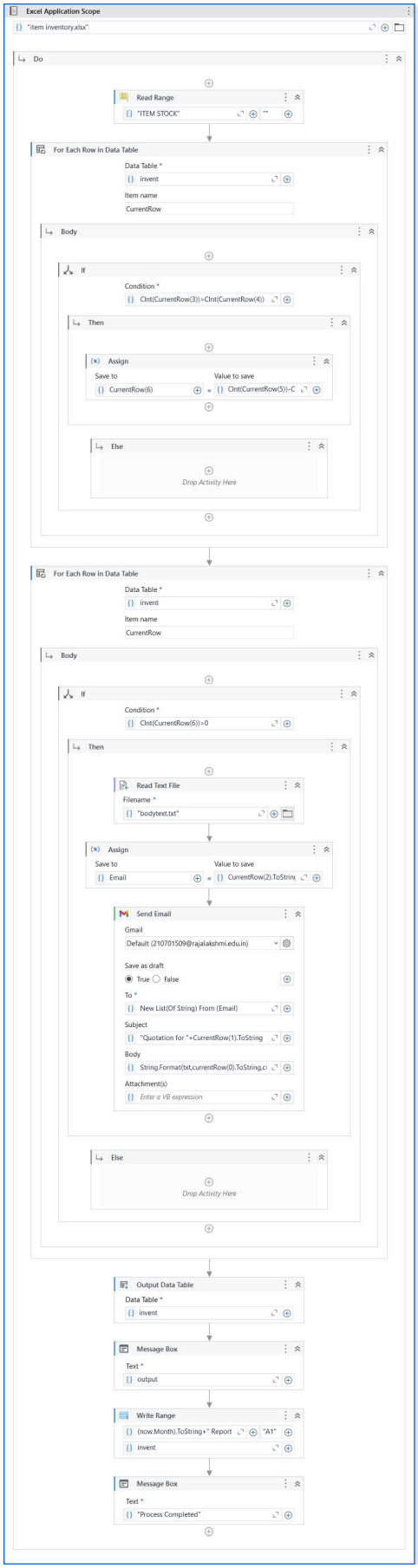
1. **AI Integration:** Exploring the integration of AI and machine learning algorithms could enable predictive analysis for inventory needs, optimizing stock levels dynamically.
2. **Enhanced Vendor Integration:** Strengthening integration with vendor systems for real-time updates on order status, inventory availability, and automated invoice processing.
3. **User Experience Refinement:** Continuous improvement in the user interface and experience to simplify interactions and reduce the learning curve for users.
4. **Expand Automation:** Identifying additional areas for automation, such as predictive maintenance, demand forecasting, or automatic reorder triggers based on consumption patterns.
5. **Security Measures:** Continuously upgrading security protocols to ensure data privacy and protection against potential cyber threats.

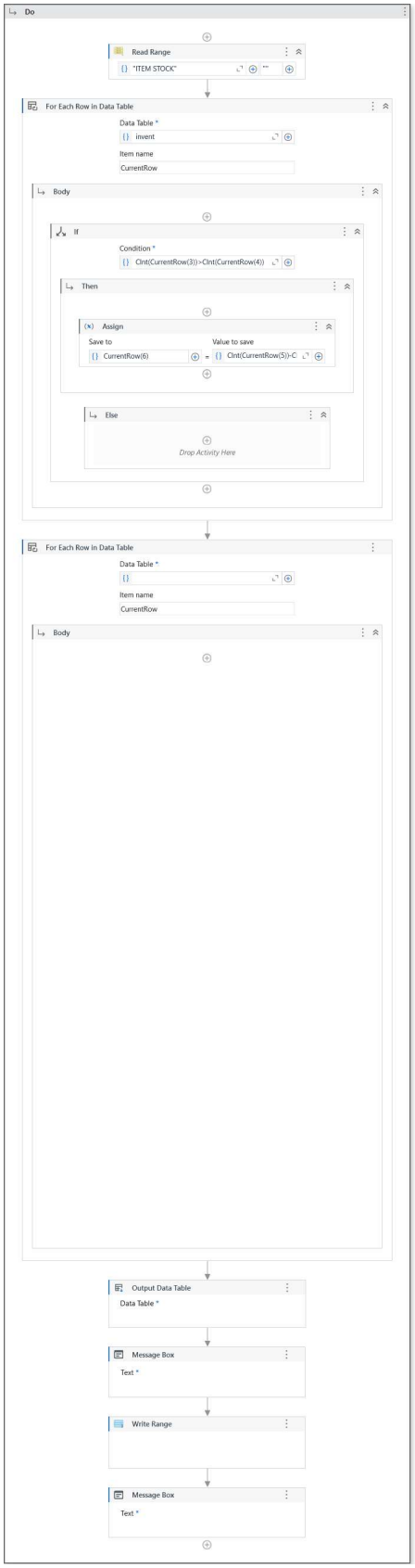
In conclusion, the UiPath-powered Inventory Order Placing System has marked a significant step towards efficient inventory management. Future enhancements aim to further optimize processes, embrace advanced technologies, and ensure sustained efficiency and competitiveness in our inventory operations.

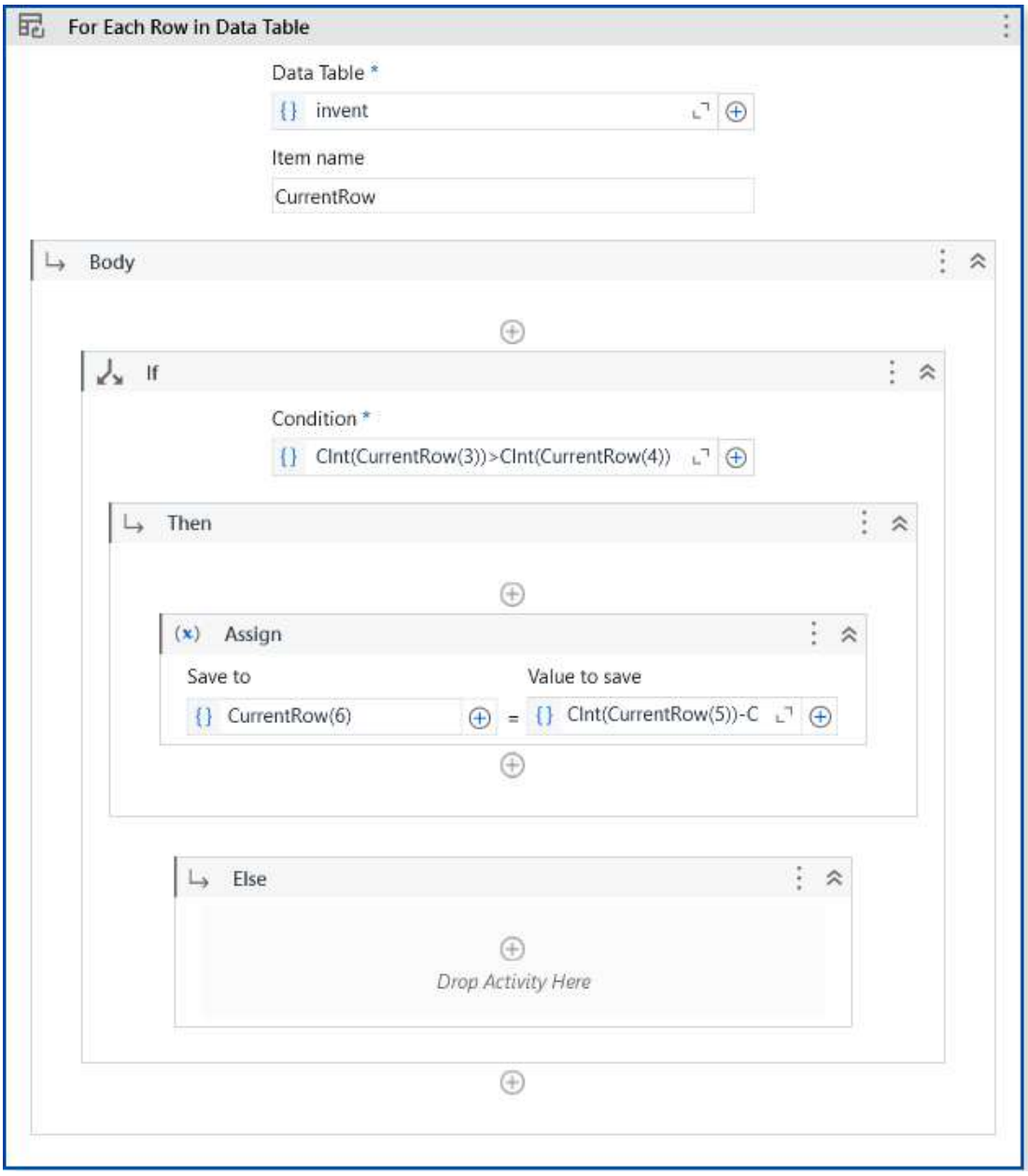
APPENDIX

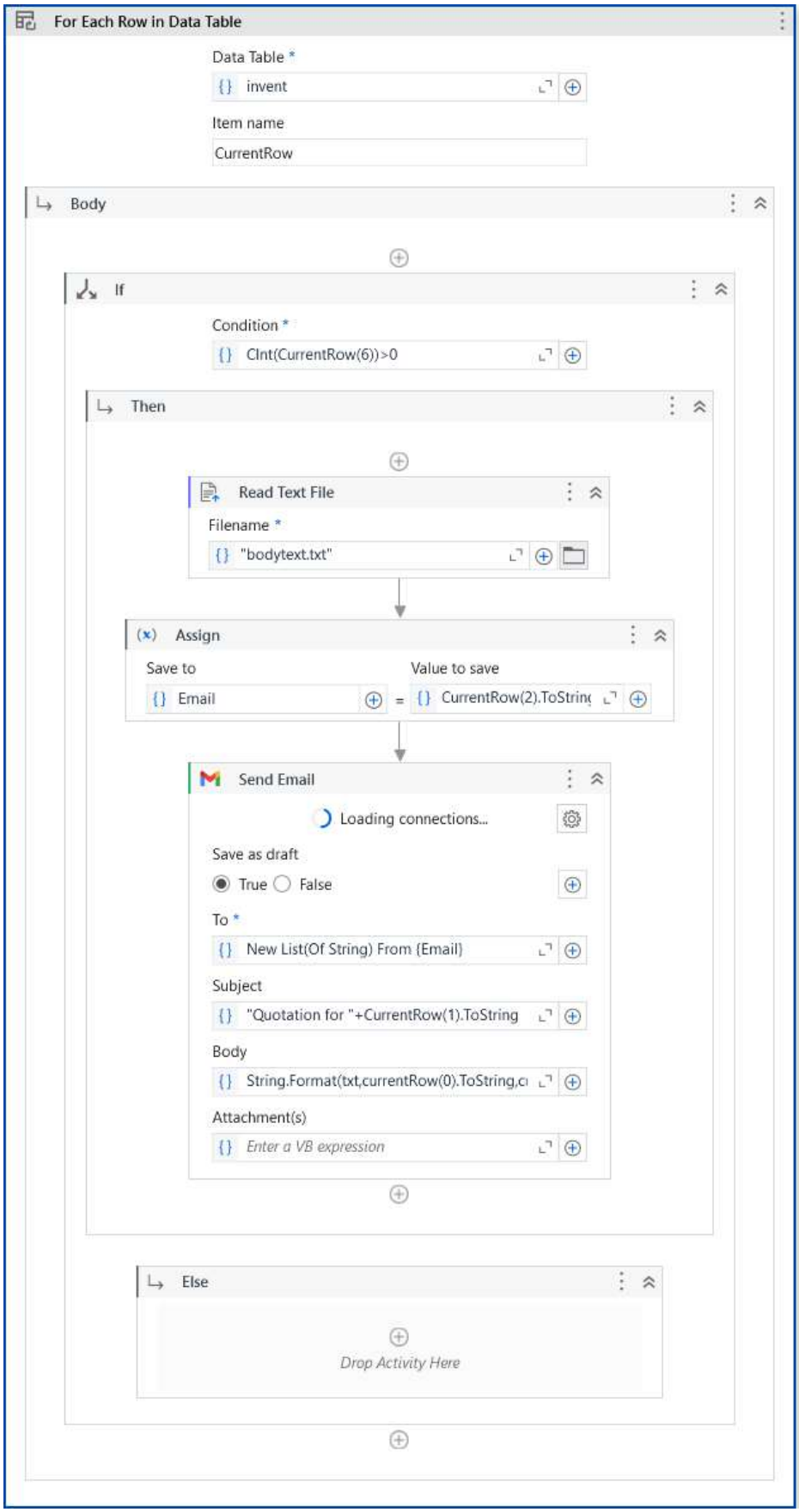
SAMPLE PROCESS

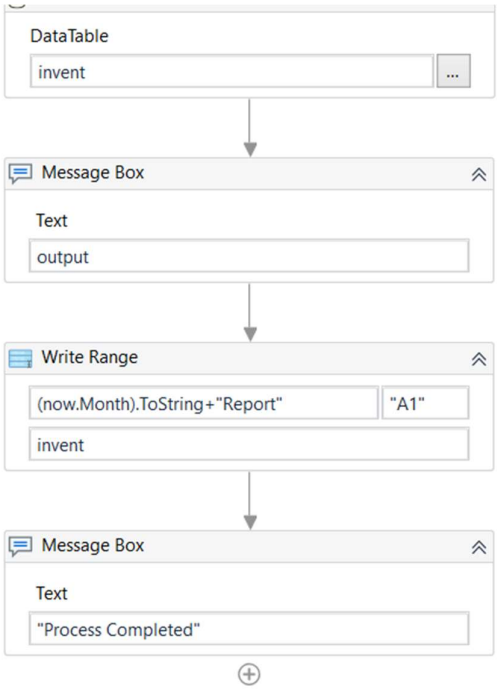



Excel Application Scope
"item inventory.xlsx"
Do
Read Range
"ITEM STOCK"
""
For Each Row in Data Table
Data Table *
invent
Item name
CurrentRow
Body
If
Condition *
CInt(CurrentRow(3))>CInt(CurrentRow(4))
Then
Assign
Save to
Value to save
CurrentRow(6)
CInt(CurrentRow(5))-C
Else
Drop Activity Here
For Each Row in Data Table
Data Table *
invent
Item name
CurrentRow
Body
If
Condition *
CInt(CurrentRow(6))>0
Then
Read Text File
Filename *
"bodytext.txt"
Assign
Save to
Value to save
Email
CurrentRow(2).ToStrin
Send Email
Gmail
Default (210701509@rajalakshmi.edu.in)
Save as draft
True
False
To *
New List(Of String) From (Email)
Subject
"Quotation for "+CurrentRow(1).ToString
Body
String.Format(txt,currentRow(0).ToString,ci
Attachment(s)
Enter a VB expression
Else
Drop Activity Here
Output Data Table
Data Table *
invent
Message Box
Text *
output
Write Range
(now.Month).ToString+" Report
"A1"
invent
Message Box
Text *
"Process Completed"


Do
Read Range
"ITEM STOCK"
For Each Row In Data Table
Data Table *
invent
Item name
CurrentRow
Body
If
Condition *
CInt(CurrentRow(3))>CInt(CurrentRow(4))
Then
Assign
Save to
Value to save
CurrentRow(6)
CInt(CurrentRow(5))-C
Else
Drop Activity Here
For Each Row In Data Table
Data Table *
Item name
CurrentRow
Body
Output Data Table
Data Table *
Message Box
Text *
Write Range
Message Box
Text *


For Each Row in Data Table
Data Table *
{} invent
Item name
CurrentRow
Body
If
Condition *
{} CInt(CurrentRow(3))>CInt(CurrentRow(4))
Then
Assign
Save to
Value to save
{} CurrentRow(6)
=
{} CInt(CurrentRow(5))-C
Else
Drop Activity Here


For Each Row in Data Table
Data Table *
invent
Item name
CurrentRow
Body
If
Condition *
CInt(CurrentRow(6))>0
Then
Read Text File
Filename *
"bodytext.txt"
Assign
Save to
Email
Value to save
CurrentRow(2).ToStrin
Send Email
Loading connections...
Save as draft
True
False
To *
New List(Of String) From {Email}
Subject
"Quotation for "+CurrentRow(1).ToString
Body
String.Format(txt,currentRow(0).ToString,c
Attachment(s)
Enter a VB expression
Else
Drop Activity Here


DataTable
invent
...
Message Box
Text
output
Write Range
(now.Month).ToString+"Report"
"A1"
invent
Message Box
Text
"Process Completed"

REFERENCES

[1] Hansen, O., Transchel, S., & Friedrich, H. (2023). Replenishment strategies for lost sales inventory systems of perishables under demand and lead time uncertainty. *European Journal of Operational Research*, *308*(2), 661-675.

[2] Kumar, A., Santra, P. K., & Mahapatra, G. S. (2023). Fractional order inventory system for time-dependent demand influenced by reliability and memory effect of promotional efforts. *Computers & Industrial Engineering*, *179*, 109191.

[3] Berling, P., Johansson, L., & Marklund, J. (2023). Controlling inventories in omni/multi-channel distribution systems with variable customer order-sizes. *Omega*, *114*, 102745.

[4] Reiman, M. I., Wan, H., & Wang, Q. (2023). Asymptotically Optimal Inventory Control for Assemble-to-Order Systems. *Stochastic Systems*, *13*(1), 128-180.

[5] Malmberg, F., & Marklund, J. (2023). Evaluation and control of inventory distribution systems with squantity-based shipment consolidation. *Naval Research Logistics (NRL)*, *70*(2), 205-227.

[6] Qi, M., Shi, Y., Qi, Y., Ma, C., Yuan, R., Wu, D., & Shen, Z. J. (2023). A practical end-to-end inventory management model with deep learning. *Management Science*, *69*(2), 759-773.

[7] Radhamani, V., & Prakash, E. (2023, May). Production inventory system with repeated demands. In *AIP Conference Proceedings* (Vol. 2718, No. 1). AIP Publishing.

[8] Al Maqbali, K. A., Joshua, V. C., Mathew, A. P., & Krishnamoorthy, A. (2023). Queueing Inventory System in Transport Problem. *Mathematics*, *11*(1), 225.