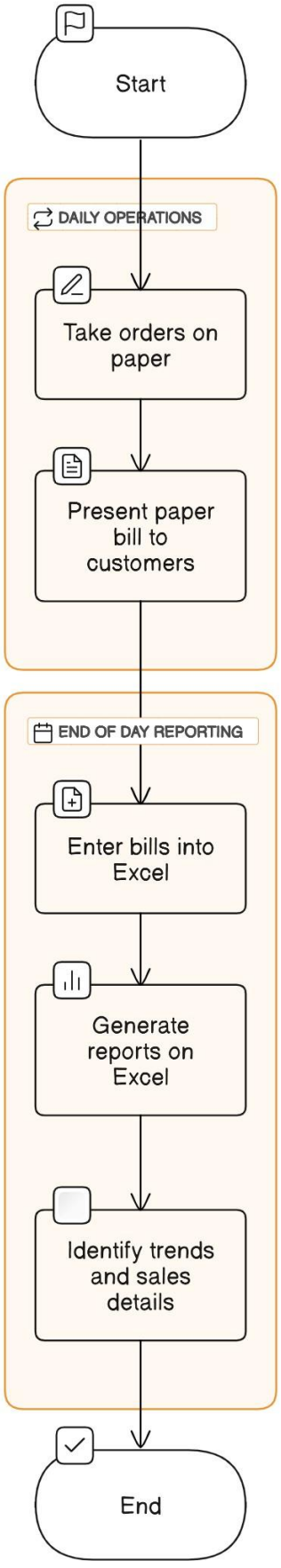
RESTAURENT MANAGEMENT SYSTEM

Submitted by
E.V. Yeswanth Kumar
Business Analyst Capstone Project
Simplilearn

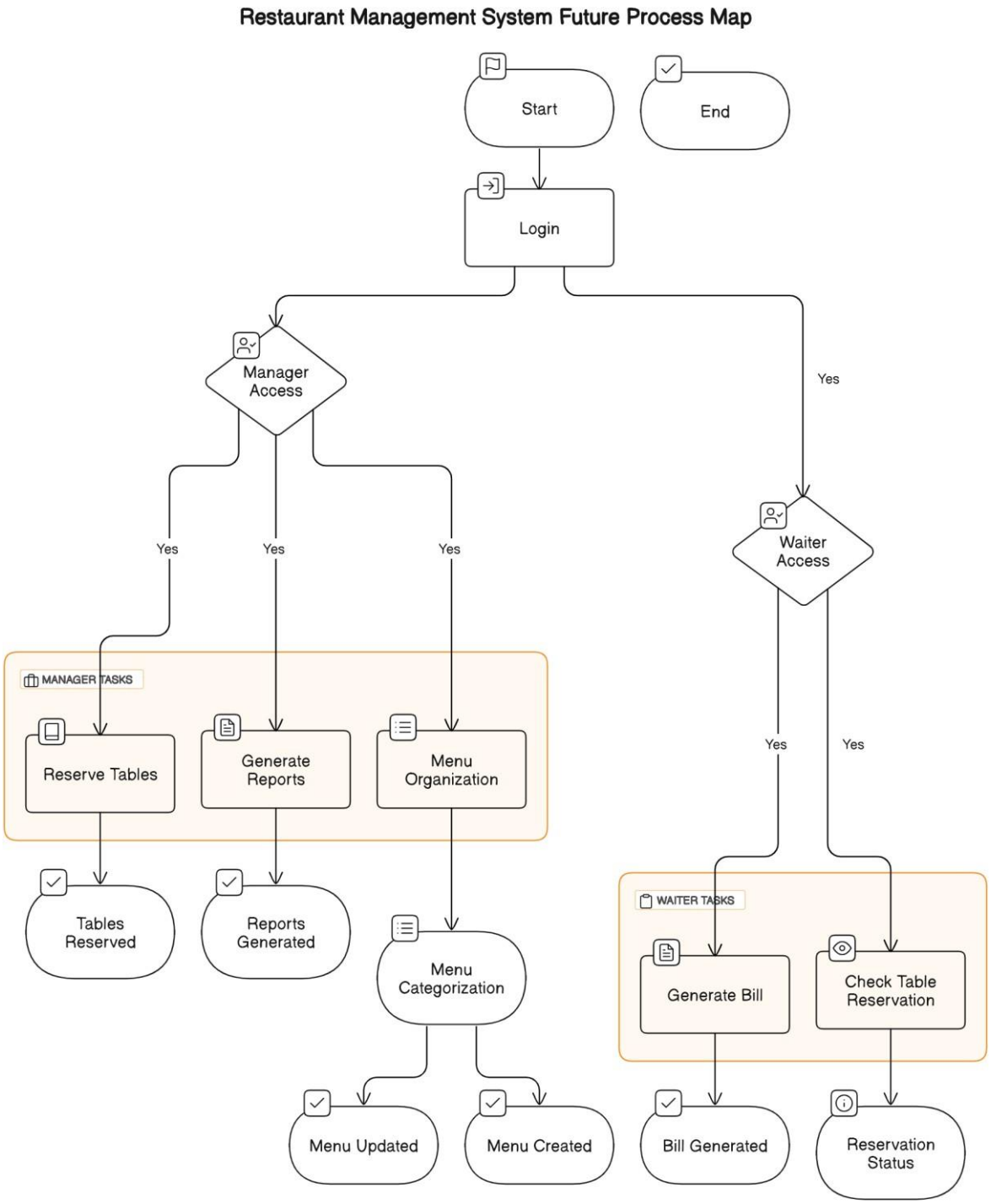
1)Identifying Stakeholders – Create a list of Stakeholders

INTERNAL STAKEHOLDERS	EXTERNAL STAKEHOLDERS
James Oliver (CEO) Managers Waiters Technical Team Developers Testers Domain SME	Customers Suppliers Regulatory Bodies Simplilearn End user

2) **As-Is Diagram for Restaurant Management System**



2)


Restaurant Management System Future Process Map
Start
End
Login
Manager Access
Yes
Waiter Access
Yes
Yes
Yes
MANAGER TASKS
Reserve Tables
Generate Reports
Menu Organization
Yes
Yes
WAITER TASKS
Tables Reserved
Reports Generated
Menu Categorization
Generate Bill
Check Table Reservation
Menu Updated
Menu Created
Bill Generated
Reservation Status

3) SCOPE OF THE PROJECT

Menu Management	Create and edit menus with classes like Starters, Soups, Main Course, Desserts, and Drinks. Allow handiest managers to adjust the menu by using adding new items, deleting existing objects, and growing new menus.
Item Search	Provide a search facility for each waiters and managers to locate objects inside the menu.
Table Reservation	Enable managers to reserve tables through the software program, stopping waiters from seating customers at reserved tables.
Bill Generation	Allow waiters to generate bills , tagging each invoice with the waiter's ID and desk number. Implement the functionality to just accept bills through coins or card via a payment gateway.
Reporting	Provide various reports for management, including : Total sales of the day by using dine-in and domestic delivery customers separately. Top 10 maximum bought dishes for the day. Total sales for weekends and months. List of dishes no longer bought inside the contemporary month. Total income across all towns and for every town.
User Authentication and Access Control	Implement a login machine for waiters, managers, and the CEO (James Oliver) with a password change facility.
Feedback Form	Provide a comments shape for customers, taking pictures information which include call, address, e-mail, date of delivery, anniversary dates, and remarks. Allow managers to manually input this facts into the system.
System Maintenance	Develop the RMS using Java for balance and simplicity of renovation over the years. Ensure that the machine is scalable and adaptable to destiny adjustments in restaurant operations.
User Interface	Design a interface for easy navigation and use by managers and waiters
Data Storage and Security	Securely store customer information and sales data to maintain privacy. Implement backup and recovery mechanisms to prevent data loss.
Integration	Integrate the RMS with existing systems or future enhancements, such as inventory management or accounting software

Training and Support	Provide training to restaurant staff on using the RMS effectively. Offer ongoing support and troubleshooting assistance post-implementation.

MAIN FEATURES THAT NEED TO BE DEVELOPED

Menu Management:	Ability for managers to create, edit, and delete menu items. Categorization of menu items into sections like Starters, Soups, Main Course, Desserts, and Drinks. Storage of menu items with their respective prices.
Search Functionality:	Allow waiters and managers to search for items in the menu.
Table Reservation:	Capability for managers to reserve tables. Integration of table layout into the system. Preventing waiters from seating customers at reserved tables.
Billing System:	Generation of bills by waiters for each table. Tagging bills to the waiter generating it and the table number. Acceptance of cash or card payments. Integration of a payment gateway.
Reporting:	Total sales of the day by dine-in and home delivery customers separately. Total consolidated sales of the day. Top 10 most sold dishes for the day. Total sales every weekend and month. List of dishes not sold in the current month. Total sales across all cities. Total sales for each city.
User Authentication and Access Control:	Login functionality for waiters, managers, and CEO James Oliver. Change password feature.
Feedback Form:	Creation of a feedback form for customers. Capture of customer details such as name, address, mobile number, email, date of birth, and anniversary dates. Manual addition of feedback details by managers into the system.
Future Enhancements:	Flexibility for future updates and enhancements. Maintenance of the system in Java to ensure minimal future code changes.

4)

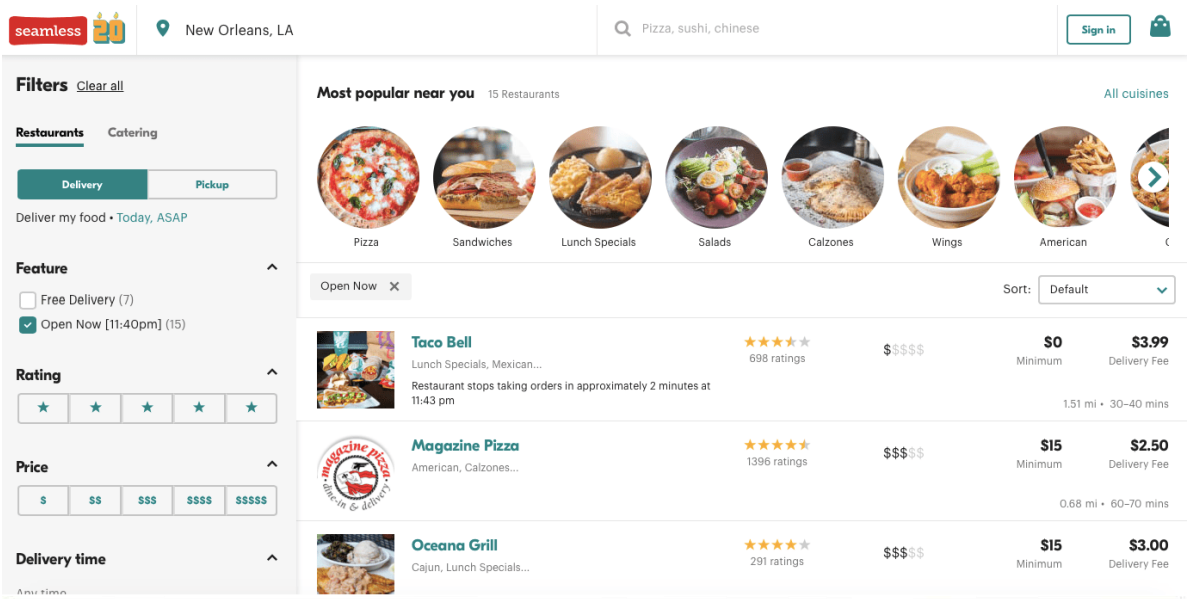
INSCOPE
1.Development of a Restaurant Management System using Java.
2.Creation of a menu with categories such as Starters, Soups, Main Course, Desserts, and Drinks.
3.Menu management functionalities for managers, including adding, deleting, and editing items, as well as creating new menus.
4.Total consolidated sales.
5.Top 10 most sold dishes.
6.Total sales on weekends.
7.Total sales every month.
8.List of dishes not sold in the current month.
9.Total sales across all cities.
10.Total sales for each city.
11.User authentication and access control for waiters, managers, and the CEO.
12.User authentication and access control for waiters, managers, and the CEO.
13.Password change functionality for users.
14.Integration of a payment gateway for cash or card payments.
15.Generation of bills for customers.
16.Implementation of a feedback form for customers to provide details and feedback, to be manually entered by managers.

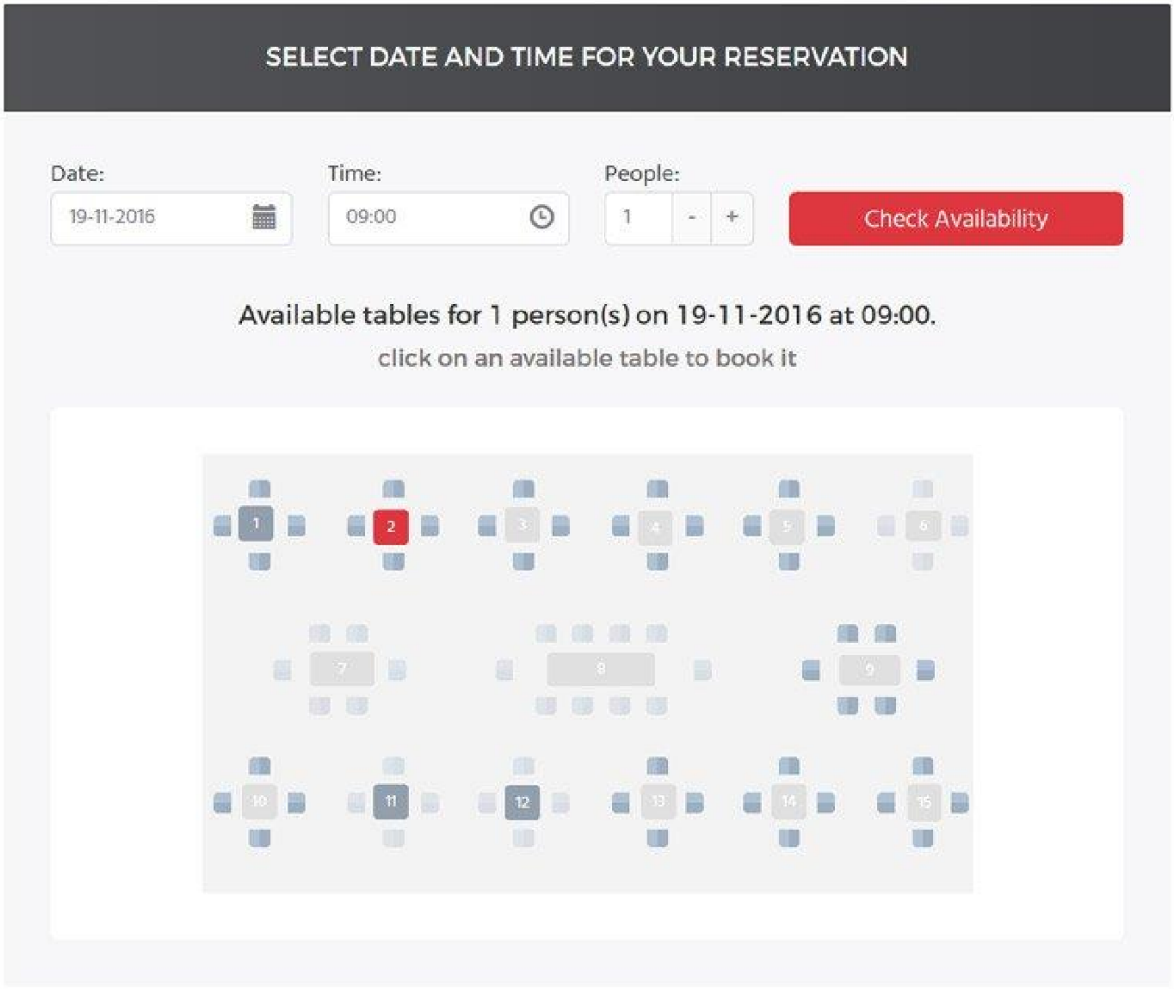
OUTSCOPE
1.Development in languages other than Java
2.Integration with third-party systems or platforms not specified.
3.Additional functionalities beyond menu management for managers, such as inventory management or employee scheduling.
4.Integration with external hardware (e.g., POS machines).
5.Automated data entry from feedback forms.
6.Integration with customer relationship management (CRM) systems.
7. Development of a mobile application for customer feedback or other purposes.
8.Implementation of loyalty programs or rewards systems.
9.Customization of menu items based on customer preferences or dietary restrictions.
10.Implementation of marketing or promotional features.
11.Support for complex analytics or forecasting beyond the specified reports.
12.Integration with social media platforms for customer engagement.
13.Development of additional paper-based forms or documentation beyond the specified feedback form.

5)

FUNCTIONAL REQUIREMNTS	NON FUNCTIONAL REQUIREMNTS
Menu Management: Managers can create, edit, and delete menu items. Menu items are categorized into Starters, Soups, Main Course, Desserts, and Drinks. Each menu item includes its name and price. **Search Facility:** Waiters and managers can search for items in the menu. **Table Reservation:** Managers can reserve tables. Waiters cannot seat anyone on reserved tables. **Billing:** Waiters generate bills per table, tagging them with the waiter's name and table number. Bills include items ordered and their respective prices. **Reporting:** Daily reports: Total sales by dine-in customers. Total sales by home delivery customers. Total consolidated sales. Top 10 most sold dishes. Periodic reports: Weekend sales. Monthly sales. List of dishes not sold in the current month. Total sales across all cities. Total sales for each city. **User Authentication and Authorization:** Login for waiters, managers, and CEO. Managers have access to menu management and table reservation. Waiters have access to bill generation. CEO may have additional privileges. **Payment Gateway:** Integration of payment gateway for cash and card transactions. Feedback Form: A feedback form to be given to every customer capturing personal details and feedback.	**Usability:** Intuitive user interface for easy navigation. Quick response time for searches and bill generation. **Reliability:** System availability during restaurant operating hours. Data integrity to ensure accurate reporting. **Security:** Secure login mechanisms to protect user accounts. Encryption of sensitive customer information. Scalability: Ability to handle increased menu items and customer transactions as the business grows. **Performance:** Efficient database management for quick retrieval of menu items and customer information. Scalable system architecture to handle concurrent users during peak hours. **Maintainability:** Well-documented codebase for easy maintenance and future updates. Modular design to facilitate addition of new features. **Compliance:** Compliance with relevant data protection regulations (e.g., GDPR, CCPA). Adherence to industry standards for payment processing security. **Interoperability:** Integration capabilities with existing systems or future third-party applications (e.g., accounting software).

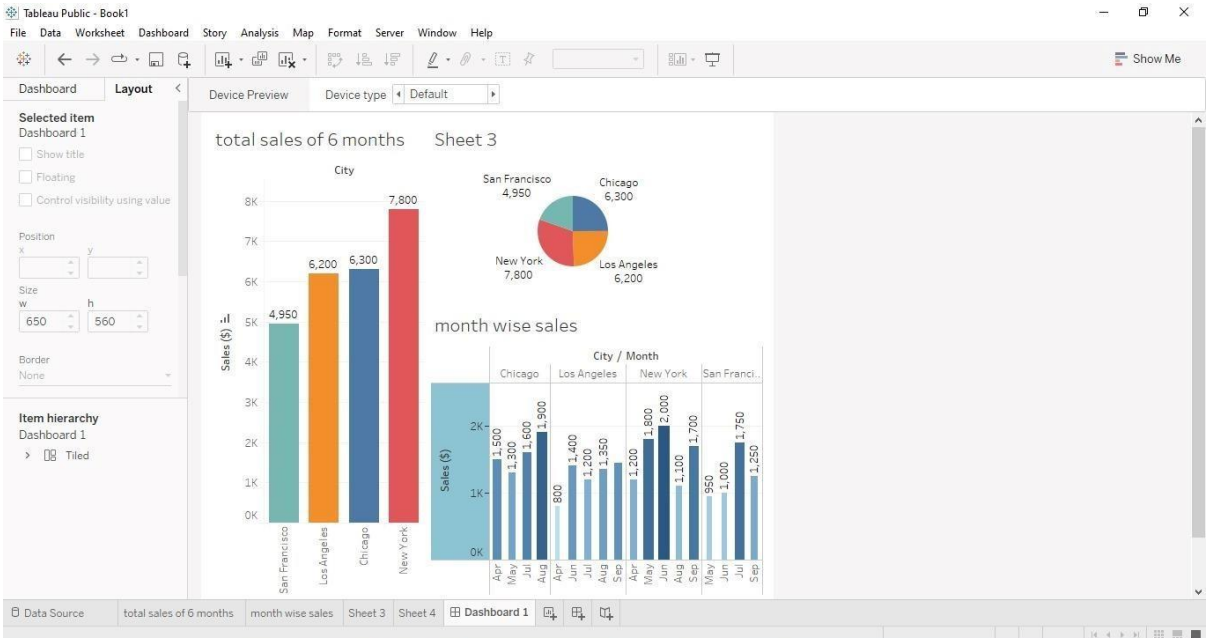
6)WIREFRAMES AND MOCKSCREEENS:



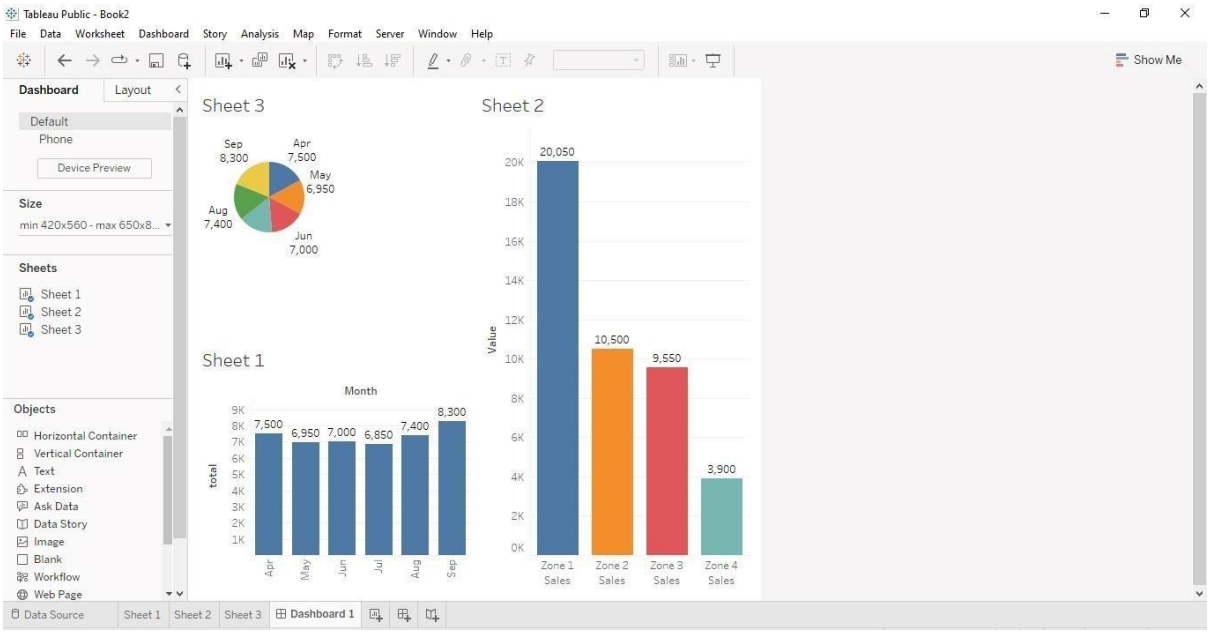


TABLUE

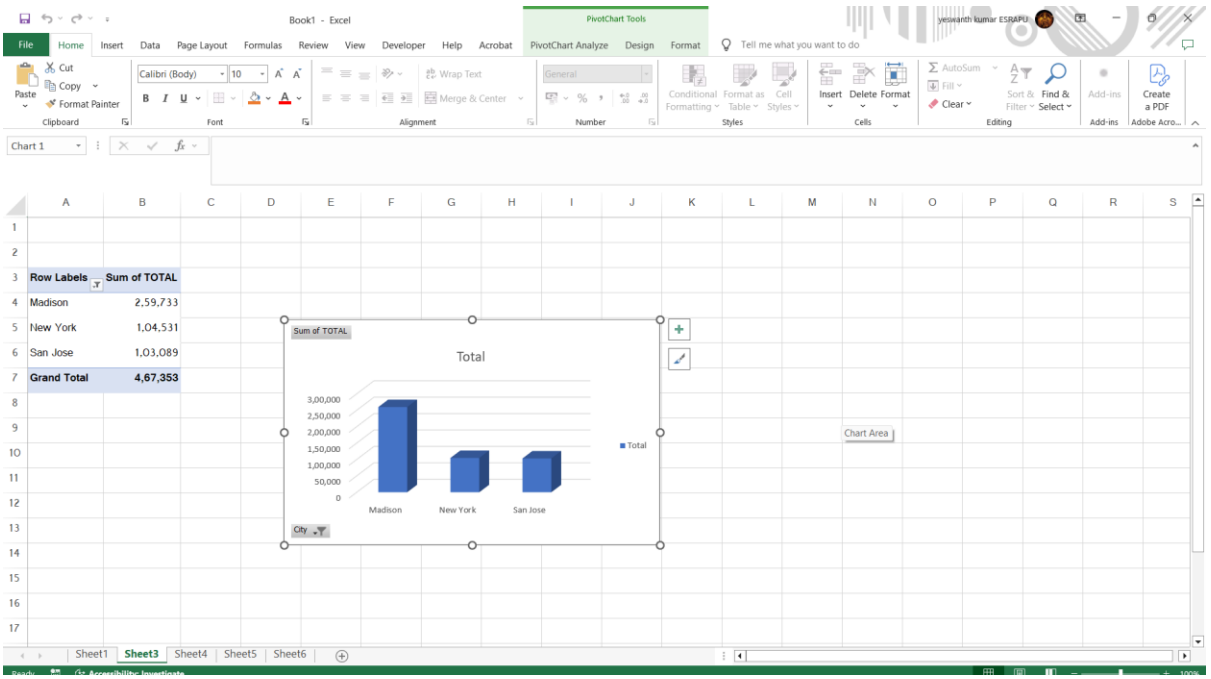
1.Create a dashboard for senior management to view sales of restaurants for the last six months. Make assumptions as appropriate and create the dashboard using your own mock data.

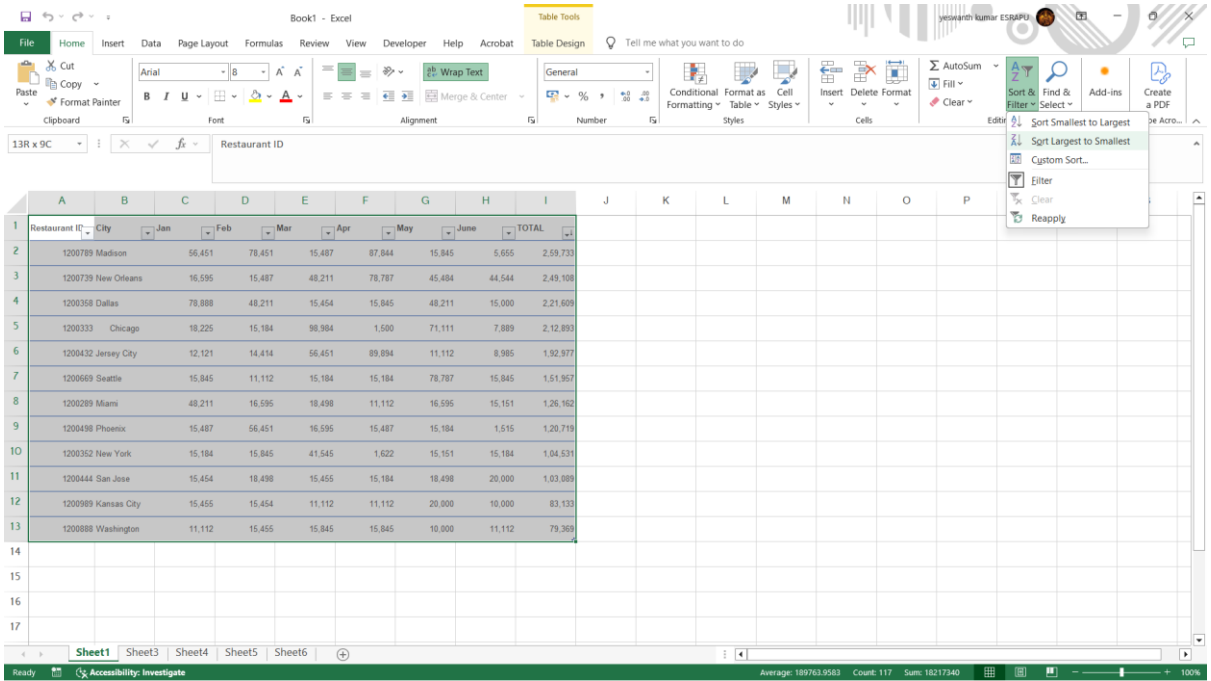


2.Create a dashboard to show which zone (Zone 1, 2, 3, or 4) has highest sales. Make assumptions as appropriate and create the dashboard using your own mock data.



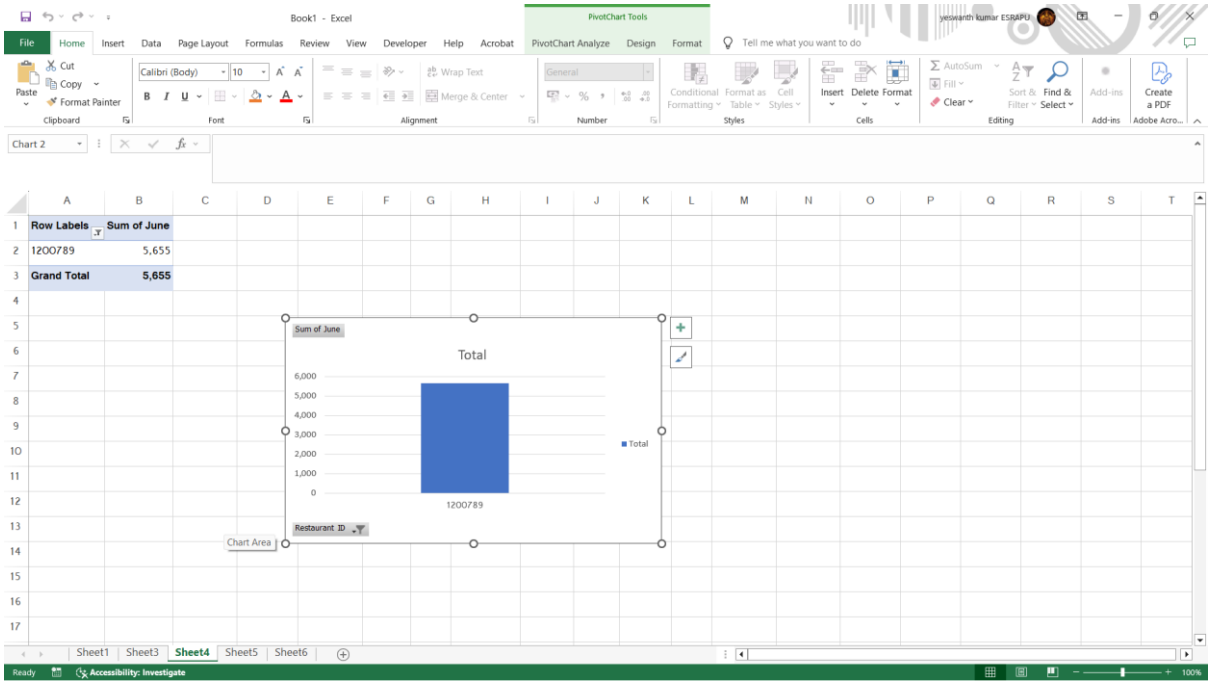
1.Create a bar graph for San Jose, Madison, and New York showing the sales. Label the chart drawn correctly so that senior management gets a clear report of sales.



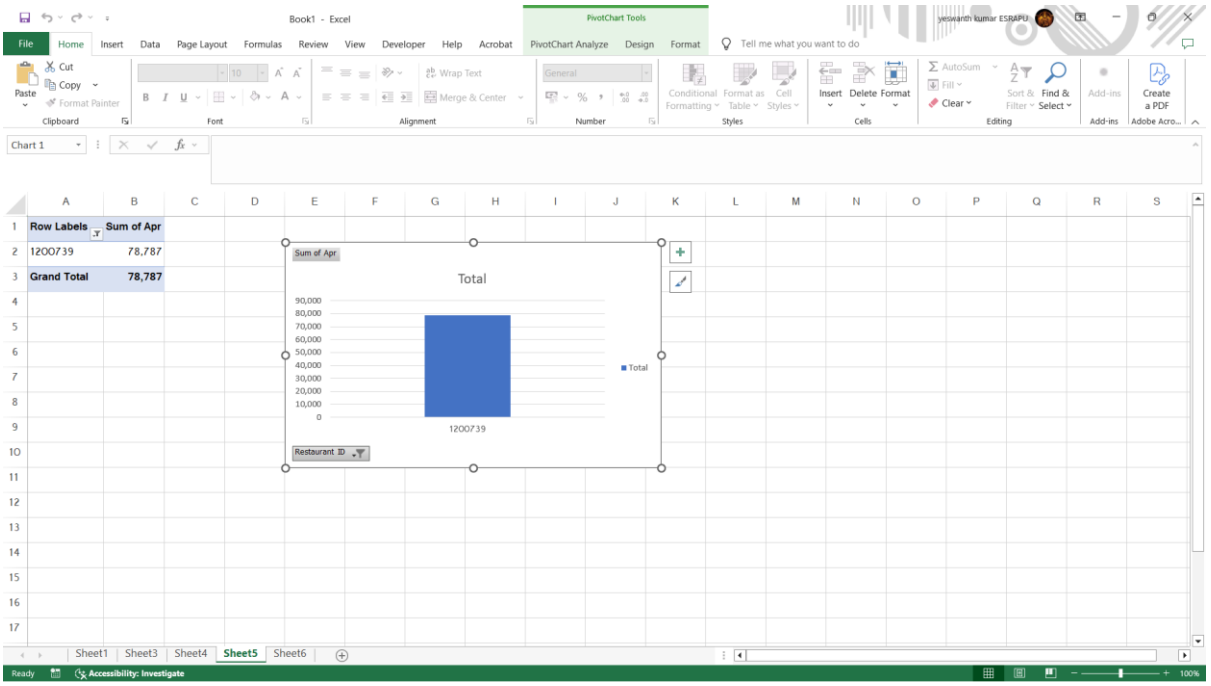


.Arrange the data above in excel in an ascending and descending order for each city.

2.In the above chart for restaurant ID 1200789, find the sales for the month of June



2.In the above chart for restaurant ID 1200739, find the sales for the month of April



2.In the above chart for restaurant ID 1200352, find the sales for the month of January