ECO HOME PRO CRM

Streamlining HVAC Excellence with Intelligent CRM Solutions

Comprehensive CRM and FSM Solution Proposal

Enhancing Efficiency and Customer Experience in the HVAC Industry

Date: 16th September 2024

Prepared by:
Anil Dhakal
Suruchi Pokhrel
Prabin Shrestha

Supervised by:
Pouria Modaresi

ABSTRACT

Imagine a bustling HVAC company struggling with outdated systems and inefficient workflows. The team struggles with managing customer interactions, scheduling service appointments, and tracking real-time data, leading to frustration and lost opportunities. Now, envision a new, all-encompassing CRM system designed to tackle these challenges head-on. This innovative solution integrates advanced features such as automated pipelines and workflows, and real-time data monitoring facilitating optimized and cost-effective customer relation management (CRM) as well as field service management. By seamlessly merging CRM and Field Service Management (FSM), it streamlines communication, enhances data accuracy, and optimizes service delivery.

The proposed target for this project is to develop an advanced CRM web-application that features customer relation management in addition to field service management. This comprehensive application will not only enhance operational efficiency but also drive business growth and improve customer satisfaction by delivering a powerful, scalable CRM solution tailored to the needs of the HVAC industry. With a focus on leveraging open-source tools for cost efficiency and flexibility, the proposed CRM system is designed to drive operational excellence, support business growth, and deliver superior user experience for all stakeholders involved. Agile software development methodologies are taken into consideration for managing software development lifecycle and CI/CD pipeline is integrated for continuous integration and deployment of the system. The system implementation will start with the development of database models, different service and features integrations and automated pipelines and workflow designs. Python and JavaScript will be used as a primary programming language for the system, Django Framework will be used for the backend services and NextJS Framework will be used for the frontend services and UI/UX designs while using PostgreSQL as a relational database and Redis as No-SQL database for high performance activities. For the production in the later phase of this project, the system will be deployed on the Azure App services while employing different other services such as Azure Database, Storage, etc. from the same cloud service provider. The expected outcome of this project is to design and develop a fully functional CRM web application that meets the day to day needs of the HVAC business with enhanced efficiency and customer experience.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Business Requirements

1.1.1 Problem Statement

Although Zoho has been our go-to platform for CRM and FSM, our team has consistently hit roadblocks trying to strike a balance between its complex customization options and ease of use. Crucial features that drive Sales and Operations are either missing or fall short of expectations. Some of the biggest pain points include frustrating user experience due to constant glitches, unreliable payment tracking, a tedious and error-prone conversion process, lack of tools to detect system anomalies, ineffective technician tracking, limited customer communication features, and an underwhelming suite for Field Service Management. These limitations are slowing us down and pushing us to rethink our approach.

1.1.2 Scope of Work

To address the limitations of our current CRM system, we aim to develop a fully customized solution tailored specifically to our business processes. This new system will prioritize user experience, streamline operations, and improve overall efficiency. The key objective is to align the CRM software with the unique requirements of our sales and operations teams, ensuring seamless integration with our existing workflows, tools, and systems. By doing so, we will eliminate current pain points, enhance communication, and offer more precise tracking of customer interactions, payments, and technician activities.

1.1.3 Proposed Solution

Our high-level approach involves building a CRM system that adapts to our specific business needs. We will design automated workflows to reduce manual tasks and errors, enabling faster and more accurate conversions. Integration with our existing tools, such as payment systems and communication platforms, will provide a unified experience. Additionally, we'll implement real-time tracking of technicians and enhanced customer engagement strategies, ensuring personalized communication at every touchpoint. By customizing key features such as reporting, anomaly detection, and Field Service Management capabilities, the solution will empower our team to work more efficiently and deliver an exceptional customer experience.

1.1.4 Targets/ Goals

- **Enhanced User Experience**: Minimize system glitches and improve ease of use to boost team productivity and user satisfaction.
- **Streamlined Conversion Process**: Simplify and automate lead-to-customer conversions, reducing errors and manual work.

- **Improved Payment Tracking**: Implement a robust system for accurately tracking customer payments and managing outstanding balances.
- **Real-Time Technician Tracking:** Enable real-time tracking of field technicians to improve service delivery and operational visibility.
- **Automated Anomaly Detection:** Develop a system to detect and report anomalies or discrepancies within the CRM and FSM processes.
- **Integrated Communication Tools:** Provide seamless communication channels with customers, improving response times and engagement.
- **Comprehensive Field Service Management (FSM) Capabilities:** Expand FSM features to include work order management, inventory tracking, and technician assignment.

1.2 Stakeholders

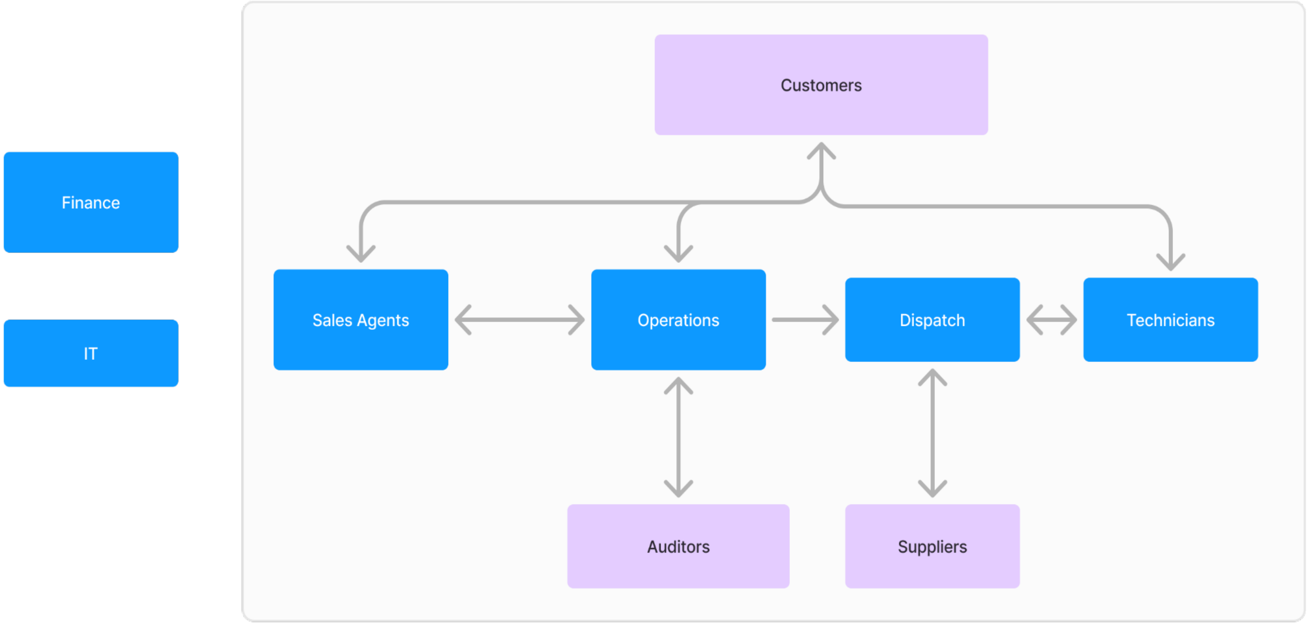


Figure 1. Internal and External Stakeholders for CRM

1.2.1 Internal Stakeholders

Operations: The Operations team plays a critical role in managing scheduling, job assignments, and customer interactions within the system. Their primary responsibilities include scheduling jobs and assigning them to technicians, handling customer paperwork, and ensuring job completion details are accurately recorded. Additionally, they communicate with Sales Agents to gather customer information and coordinate with Dispatch to ensure proper equipment and job scheduling. Operations teams rely heavily on the system’s scheduling tools, customer interaction logs, and job progress tracking features to ensure smooth operations.

Dispatch: Dispatch teams oversee technician assignments and ensure that jobs are dispatched in a timely manner. They are responsible for assigning technicians based on location, availability, and specific skill sets required for each job. Furthermore, Dispatch ensures that all necessary equipment is dispatched on time and that technicians can access it without delay. They rely on the system’s technician scheduling tools, GPS tracking, and equipment management modules to optimize technician routes and manage job logistics efficiently.

Technicians: Technicians perform on-site jobs and update the system in real time to report job status and completion. They are responsible for completing assigned jobs and reporting any issues encountered during the process. After finishing a job, they update the system with job completion details, customer feedback, and any follow-up requirements. To streamline this, technicians use mobile access to update job statuses, log service details, and maintain communication with Dispatch, ensuring that job details are accurately recorded and addressed promptly.

Sales Agents: Sales Agents manage customer relationships and are responsible for generating sales and providing customer information to both Operations and Dispatch teams. Their tasks include tracking customer interactions, following up on leads, and managing sales opportunities. The system serves as a vital tool for Sales Agents, allowing them to efficiently manage leads, track customer data, and ensure smooth communication of customer requirements to Operations and Dispatch.

Finance: The Finance team handles billing, invoicing, and financial tracking for all jobs completed within the system. Their responsibilities include processing payments, generating invoices, and ensuring that all financial records are up to date. They work closely with Operations to track job costs and profitability. The Finance team utilizes financial modules within the system to manage billing, invoicing, and expense tracking, ensuring that all transactions are recorded and monitored for accuracy.

IT: IT is responsible for maintaining the system and ensuring it runs efficiently and securely. Their responsibilities include providing support for system issues, managing updates and integrations, and ensuring data security and system reliability. IT oversees system administration, manages integrations with other tools, and provides user support across departments, ensuring the system remains functional and aligned with the organization's needs.

1.2.2 External Stakeholders

Customers: Customers interact with the system primarily to receive services, book appointments, provide feedback, and receive updates on job status. These interactions occur through customer portals or direct communication with service representatives. The system helps ensure that customers are kept informed about their service appointments and provides an avenue for them to share their feedback.

Suppliers: Suppliers provide the equipment and materials necessary for job completion. Their interaction with the system typically involves integration with inventory management to track equipment availability and ensure timely ordering of new supplies as needed. This integration ensures that Dispatch and Operations teams can manage job requirements effectively without delays due to supply shortages.

Auditors: Auditors ensure that operations comply with industry regulations and review system processes for accuracy and efficiency. They interact with the system by accessing records, job logs, financial transactions, and compliance documentation. The system’s robust tracking and reporting features provide auditors with the transparency needed to review operations and financial transactions, ensuring compliance and operational accuracy.

2. Methodology

2.1 Agile Software Development

Agile software development refers to software development methodologies centered around the idea of iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

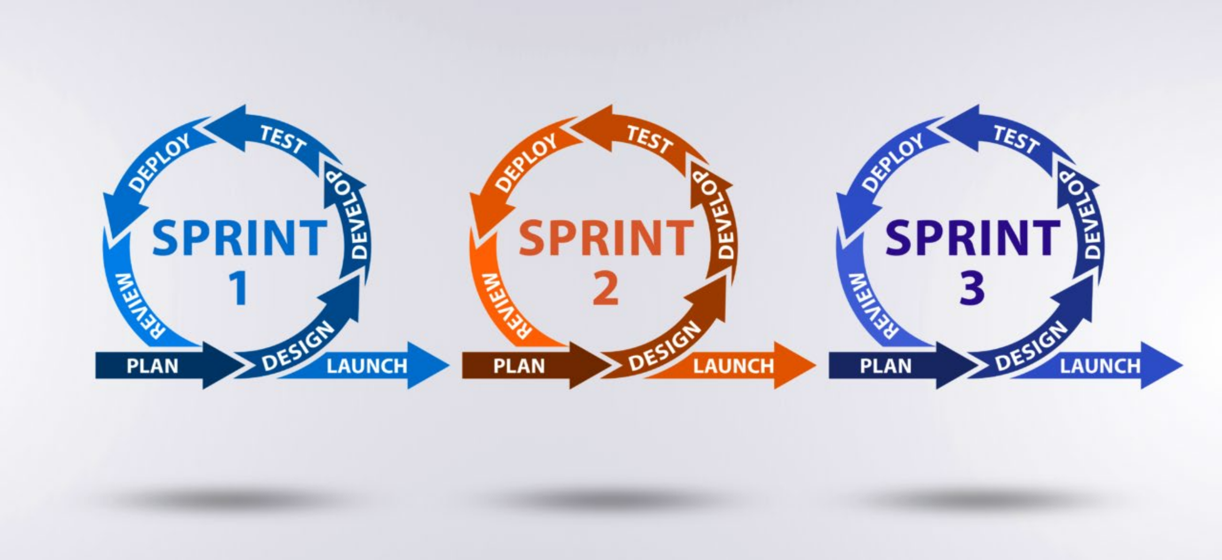


Figure 2 Agile Project Methodology

In an agile development method, the project is broken down into several modules, and each of the modules are scheduled for multiple development cycles called a sprint. Each sprint is usually a week or two long and focuses on the development of a specific set of features. It undergoes planning, design, development, testing, deployment and review process until the desired requirements are met. Each of the tasks can be assigned with a numerical indicator called story point that can help estimate the working hours to complete the task. The story point is based on the time required to complete the task and its complexity.

2.1.1 Sprint 1: Local Development

- **End Time: 15th December**

This phase focuses on building and testing all core modules on a local environment, ensuring the foundational aspects of the system are functioning correctly. The goal is to complete all critical features by the end of this phase.

- **All Modules**: This includes the primary modules such as customer management, sales order processing, work order generation, technician assignment, and product inventory management. Each module will be tested locally for stability and functionality.

- **Facebook Integration**: Integration with Facebook will allow customer interactions through social media, enabling the system to receive leads, inquiries, and customer messages directly from Facebook. This will streamline social media engagement and allow quick response to customer queries.
- **Google Calendar Integration**: This will allow seamless appointment scheduling by syncing work orders and technician schedules with Google Calendar. This ensures all appointments are automatically updated in real time, preventing scheduling conflicts.
- **Square Integration**: Square will be used for payment processing, allowing the system to manage transactions, invoicing, and payment tracking for completed work orders. Square integration ensures secure and streamlined payment handling.
- **Web CRM App (Local)**: The web-based CRM app will be built and hosted locally in this phase. Users can manage customer data, work orders, and sales orders from a local environment.
- **Email and SMS Communication**: The system will integrate email and SMS functionalities for outbound communication, including appointment confirmations, payment reminders, and follow-ups with customers.
- **Outbound Calls**: The system will also enable outbound calls to customers, allowing service teams to directly contact customers to confirm appointments or provide updates.

2.1.2 Sprint 2: Cloud Services Integration

Once the core system is stable and functional locally, this phase transitions the system to cloud services, enabling broader access and scalability.

- **Cloud Services Integration**: This involves migrating the system to a cloud environment for increased scalability, reliability, and remote access. Cloud services will also allow seamless backups, data security, and multi-location access.
- **Staging Server**: A staging environment will be set up in the cloud to test the system in conditions that mirror the production environment. This ensures that the system is stable and any bugs are identified and fixed before production deployment.
- **Internal QA**: This phase will include rigorous internal quality assurance (QA) testing. QA will focus on identifying bugs, validating features, and ensuring that the system operates according to the project’s specifications.
- **Additional Mandatory Integrations**: Integrations with other tools and platforms (beyond Facebook, Google Calendar, and Square) that are necessary for full system functionality will be added. This could include additional payment gateways, third-party service platforms, or additional communication tools like Slack.
- **Additional Mandatory Features**: Any core features that were not included in Phase 1 or new requirements discovered during development will be added in this phase. This could include advanced reporting, role-based access control, or extended customer segmentation.

2.1.3 Sprint 3: Deployment in Production

After testing is complete, the system is deployed to a live production environment, making it available for real users.

- **Deployment in Production**: The system will be moved to a live production environment where actual users (e.g., technicians, sales agents, and managers) can interact with the system in real-world conditions. All cloud services will be operational, and real-time data will be used.
- **User Acceptance Testing (UAT)**: UAT involves actual end-users interacting with the system to ensure that it meets their needs and works as expected in their daily operations. Their feedback will help validate whether the system is ready for widespread use.
- **User Feedback and Improvements**: Based on user feedback during UAT, the system will undergo further refinements and improvements. These could involve fixing minor bugs, tweaking user interfaces, or adding features to enhance user experience and meet business needs.

2.2 CI/CD Pipeline

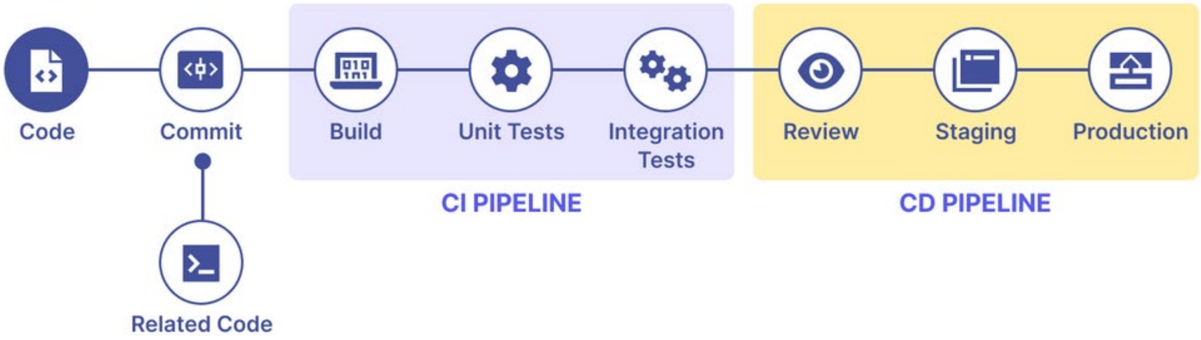


Figure 3 CI/CD Pipeline

A CI/CD pipeline is an automated process utilized by software development teams to streamline the creation, testing and deployment of applications. "CI" represents continuous integration, where developers can only merge code changes into a central repository once all the automated tests have passed, allowing early detection of issues. "CD" refers to continuous deployment or continuous delivery, which automates the application's release to its intended environment, ensuring that it is readily available to users. This pipeline is vital for teams aiming to improve software quality and speed up delivery through regular, reliable updates.

Integrating a CI/CD pipeline significantly reduces the risk of errors in the deployment process. Automating builds and tests ensures that bugs are caught early and fixed promptly, maintaining high-quality software. Additional features such as triggering build notifications can help track the deployment status of the system.

2.3 Market Analysis

2.3.1 Customer Relationship Management Software Applications

Table 1 CRM Software Comparison

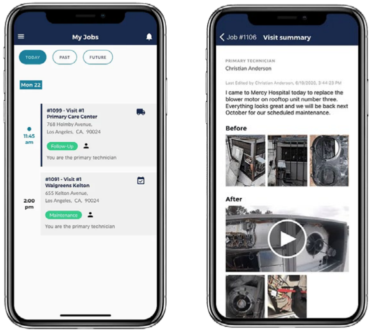
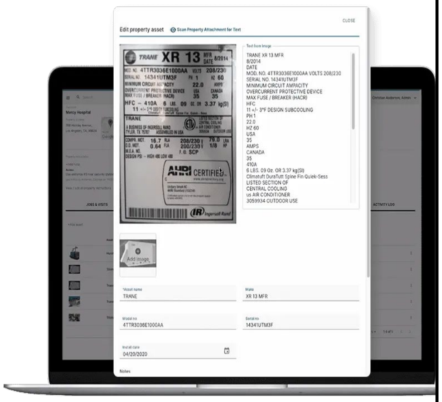
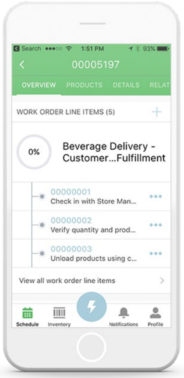
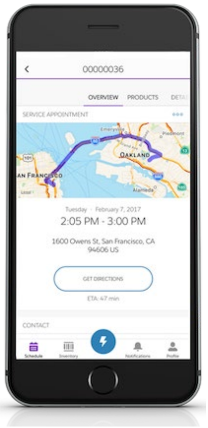
CRM	Zoho	HubSpot	mondayCRM	Pipedrive
Pricing (CAD)	Standard: 19 Professional: 31 Enterprise: 50 Ultimate: 65 (/user/month)		Basic: 18 Standard:23 Pro: 39 Enterprise: (/seat/month)	USD Essential:14 Advanced: 34 Professional: 49 Power: 64 Enterprise: 99
Modules	Leads Contacts Accounts Deals	Contact Company Deals Tickets		Leads Deals Projects
Deals Pipeline	>Qualification >Needs Analysis >Value Proposition >Identify Decision Makers >Proposal/Price Quote >Negotiation/Review >Closed Won >Closed Lost >Closed Lost to Competition		>Lead >Qualified >Proposal >Negotiation >Contact sent >Closed	>Qualified >Contact Made >Demo Scheduled >Proposal Made >Negotiation Started
Features	Blueprints Workflows Rules	Website Pages Marketing ads/campaigns		Automatic Lead/Deal Assignment

2.3.2 Field Service Management Software Applications

Software solutions that help companies manage their field service operations in a more efficient and effective manner.

- Real-time tracking of field service technicians
- Automates scheduling, dispatching, tracking and reporting reports.

Table 2 FSM Software Apps

FSM	BuildOPS	SalesForce
Screens	Technician Field Service App  My Jobs TODAY PAST FUTURE Mon 22 11:45 am 2:00 pm Follow-Up Maintenance Job #1106 Visit summary Before After #1099 - Visit #1 Primary Care Center 768 Holmby Avenue, Los Angeles, CA, 90024 You are the primary technician #1091 - Visit #1 Walgreens Kelton 655 Kelton Avenue, Los Angeles, CA, 90024 You are the primary technician PRIMARY TECHNICIAN Christian Anderson Last Edited by Christian Anderson, 6/10/2020, 3:44:23 PM I came to Mercy Hospital today to replace the blower motor on rooftop unit number three. Everything looks great and we will be back next October for our scheduled maintenance.  Edit property asset Scan Property Attachment for Text CLOSE Text from image Add Image Notes TRANE XR 13 MFR 8/2014 DATE MOD. NO. 4TTR3036E1000AA VOLTS 208/230 SERIAL NO. 14341UTM3F MINIMUM CIRCUIT AMPACITY OVERCURRENT PROTECTIVE DEVICE MAX FUSE / BREAKER (HACR) HFC 11 +/- 3°F DESIGN SUBCOOLING PH 1 22.0 HZ 60 USA 35 AMPS CANADA 35 410A 6 LBS. 09 Oz. OR 3.37 kg(SI) Climatuft DuraTutt Spine Fin Quiek-Sess LISTED SECTION OF CENTRAL COOLING us AIR CONDITIONER 3059934 OUTDOOR USE *Asset name TRANE Make XR 13 MFR Model no 4TTR3036E1000AA Serial no 14341UTM3F Install date 04/20/2020	 Search 1:51 PM 93% OVERVIEW PRODUCTS DETAILS RELAT 0% Schedule Inventory Notifications Profile 00005197 WORK ORDER LINE ITEMS (5) Beverage Delivery - Customer...Fulfillment 00000001 Check in with Store Man... 00000002 Verify quantity and prod... 00000003 Unload products using c... View all work order line items  OVERVIEW PRODUCTS DETAI SERVICE APPOINTMENT Emeryville Piedmont OAKLAND AN FRANCISCO Alameda GET DIRECTIONS CONTACT Schedule Inventory Notifications Profile 00000036 Tuesday · February 7, 2017 2:05 PM - 3:00 PM 1600 Owens St, San Francisco, CA 94606 US ETA: 47 min
	Mobile Invoicing, Quotes and Proposals ERP Accounting Software	Field Service Mobile Workers - Real-time view of data - Review schedules - Recommend routes - Ensure rights tools and equipment

3. System Design and Implementation

3.1 Modules

1. Users - Profile, Groups, Roles
2. Leads
3. Deals
4. Customers
5. Tickets
6. Estimates/Quotes
7. Sales Orders
8. Work Orders
9. Purchase Orders
10. Invoices
11. Payments
12. Products
13. Inventory Stocks
14. Serial Numbers
15. Warehouses
16. Address
17. Companies
18. Vendors
19. Analytics/Dashboard
20. Appointments/Meetings
21. Notes
22. Emails
23. Calls

3.2 Entity Relationship (ER) Diagrams - Database Model

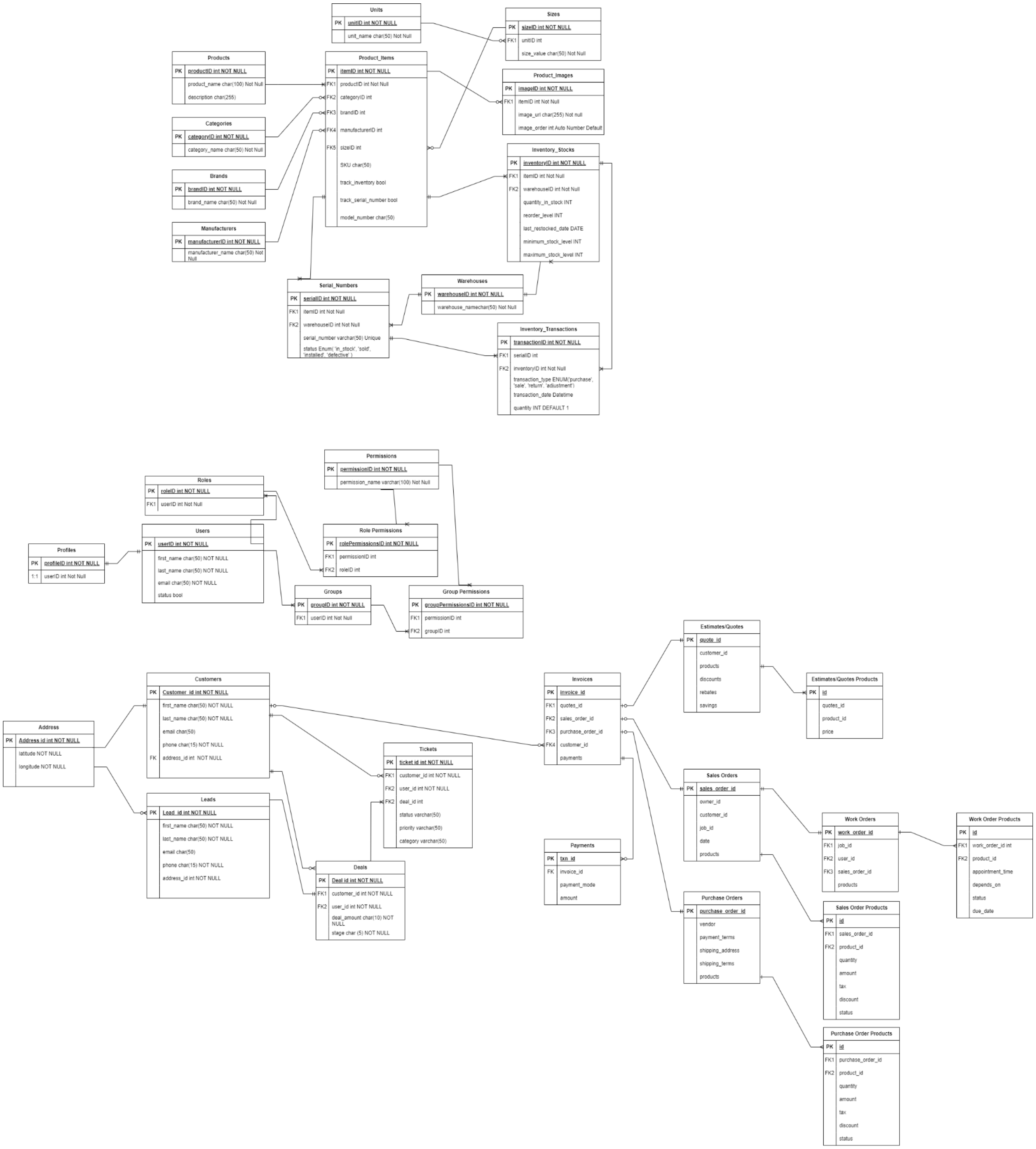


Figure 4. Overall System Data Model

This is an overview of the entire system's data model, which is composed of four key components: Users & Permissions, Products & Inventory Management, General Leads, Deals, & Customers, and Orders & Payments. Each component represents a distinct area of the system's functionality. The Users & Permissions model defines how user roles and access control are managed, ensuring secure operations. The Products & Inventory Management model handles the cataloging, tracking, and availability of products. The Leads, Deals, & Customers model captures customer interactions, lead management, and sales processes. Finally, the Orders & Payments model manages transaction processing, from order placement to payment completion. We will explore each of these models in detail in the following sections.

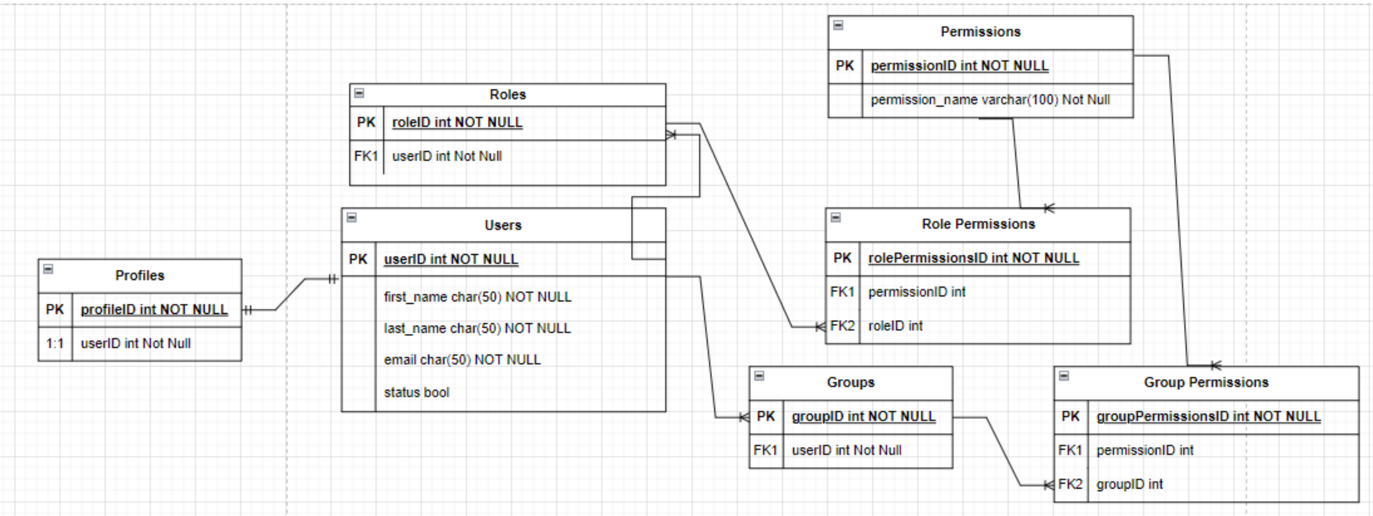


Figure 5. User, Profile, Groups & Roles Data Model

In our system, the management of users, profiles, groups, and roles is fundamental and critical. Each user contains essential information and is associated with a profile that holds additional details such as date of birth, profile image, and address. Groups are collections of individuals with similar responsibilities within a company and represent access levels based on departments. For instance, the IT Group includes members from the IT department, while the Sales Group comprises individuals from the Sales department. Within each group, members may hold various roles with differing responsibilities in addition to their shared duties. The roles data model in our system captures these distinctions, such as the "Team Leader" role within the Sales Group.

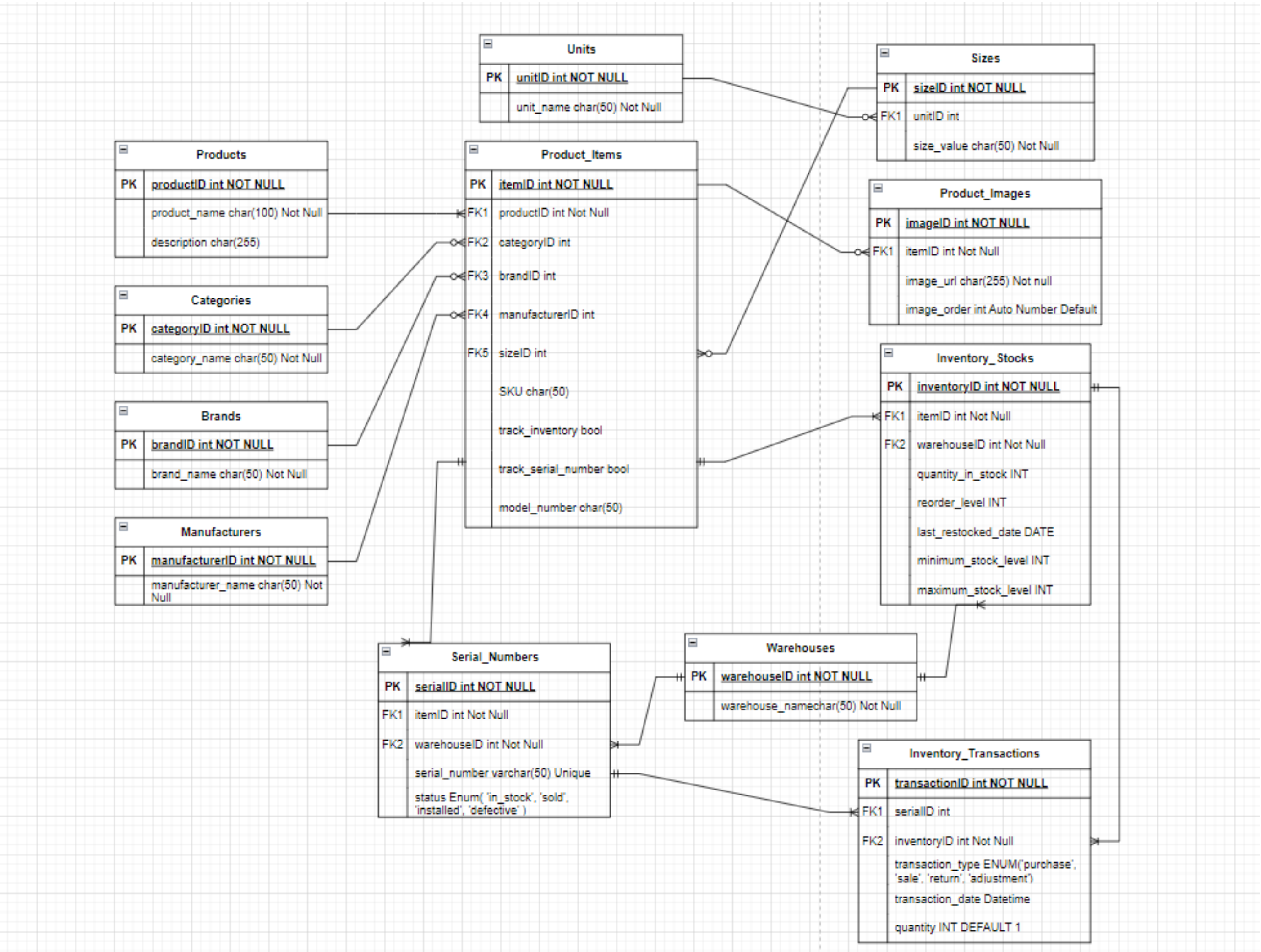

Units
PK unitID int NOT NULL
unit_name char(50) Not Null
Sizes
PK sizeID int NOT NULL
FK1 unitID int
size_value char(50) Not Null
Products
PK productID int NOT NULL
product_name char(100) Not Null
description char(255)
Product_Items
PK itemID int NOT NULL
FK1 productID int Not Null
FK2 categoryID int
FK3 brandID int
FK4 manufacturerID int
FK5 sizeID int
SKU char(50)
track_inventory bool
track_serial_number bool
model_number char(50)
Product_Images
PK imageID int NOT NULL
FK1 itemID int Not Null
image_url char(255) Not null
image_order int Auto Number Default
Categories
PK categoryID int NOT NULL
category_name char(50) Not Null
Brands
PK brandID int NOT NULL
brand_name char(50) Not Null
Manufacturers
PK manufacturerID int NOT NULL
manufacturer_name char(50) Not Null
Inventory_ Stocks
PK inventoryID int NOT NULL
FK1 itemID int Not Null
FK2 warehouseID int Not Null
quantity_in_stock INT
reorder_level INT
last_restocked_date DATE
minimum_stock_level INT
maximum_stock_level INT
Serial_Numbers
PK serialID int NOT NULL
FK1 itemID int Not Null
FK2 warehouseID int Not Null
serial_number varchar(50) Unique
status Enum('in_stock', 'sold', 'installed', 'defective')
Warehouses
PK warehouseID int NOT NULL
warehouse_namechar(50) Not Null
Inventory_Transactions
PK transactionID int NOT NULL
FK1 serialID int
FK2 inventoryID int Not Null
transaction_type ENUM('purchase', 'sale', 'return', 'adjustment')
transaction_date Datetime
quantity INT DEFAULT 1

Figure 6. Products & Inventory Management Data Model

The products and inventory management data model depicted in the diagram is a robust framework suitable for both e-commerce and CRM systems. It effectively captures various scenarios by maintaining separate tables for product attributes such as category, brand, manufacturer, and sizes. These tables are combined to form a comprehensive product item table. Products can be classified into two types: services and products. Products are further categorized as serialized or non-serialized. The inventory stocks table tracks stock levels for product items, while the serial numbers table monitors serial numbers for serialized product items, complementing the inventory stocks. All transactions involving the movement of products in and out of the warehouse are recorded in the inventory transaction table.

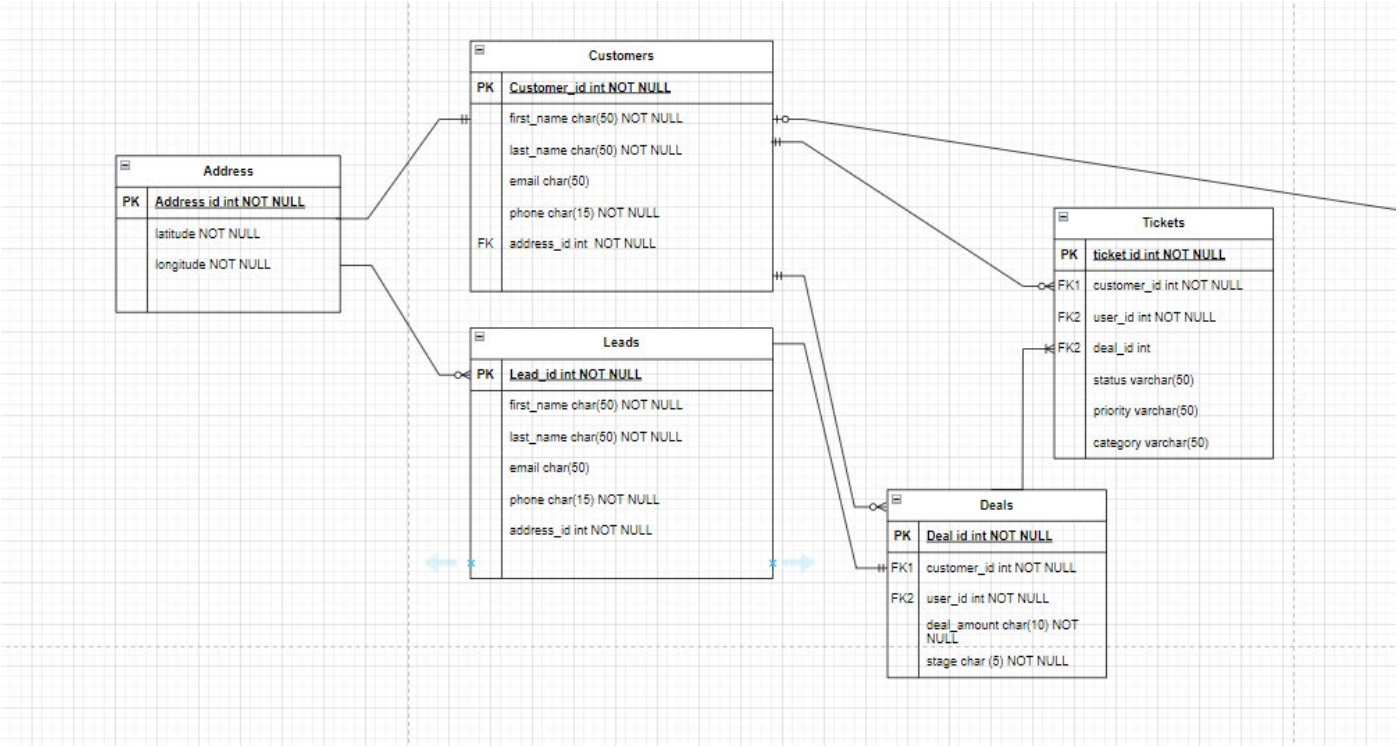


Figure 7. General Leads, Deals & Customers Data Model

The data model illustrated in these figures outlines the core modules utilized in our CRM system, including Leads, Deals, and Customers, among others. Each module is designed to integrate seamlessly with customer information and addresses as needed. Furthermore, these modules are interconnected with other essential components of the CRM, such as Notes, SMS, Calls, and Meetings.

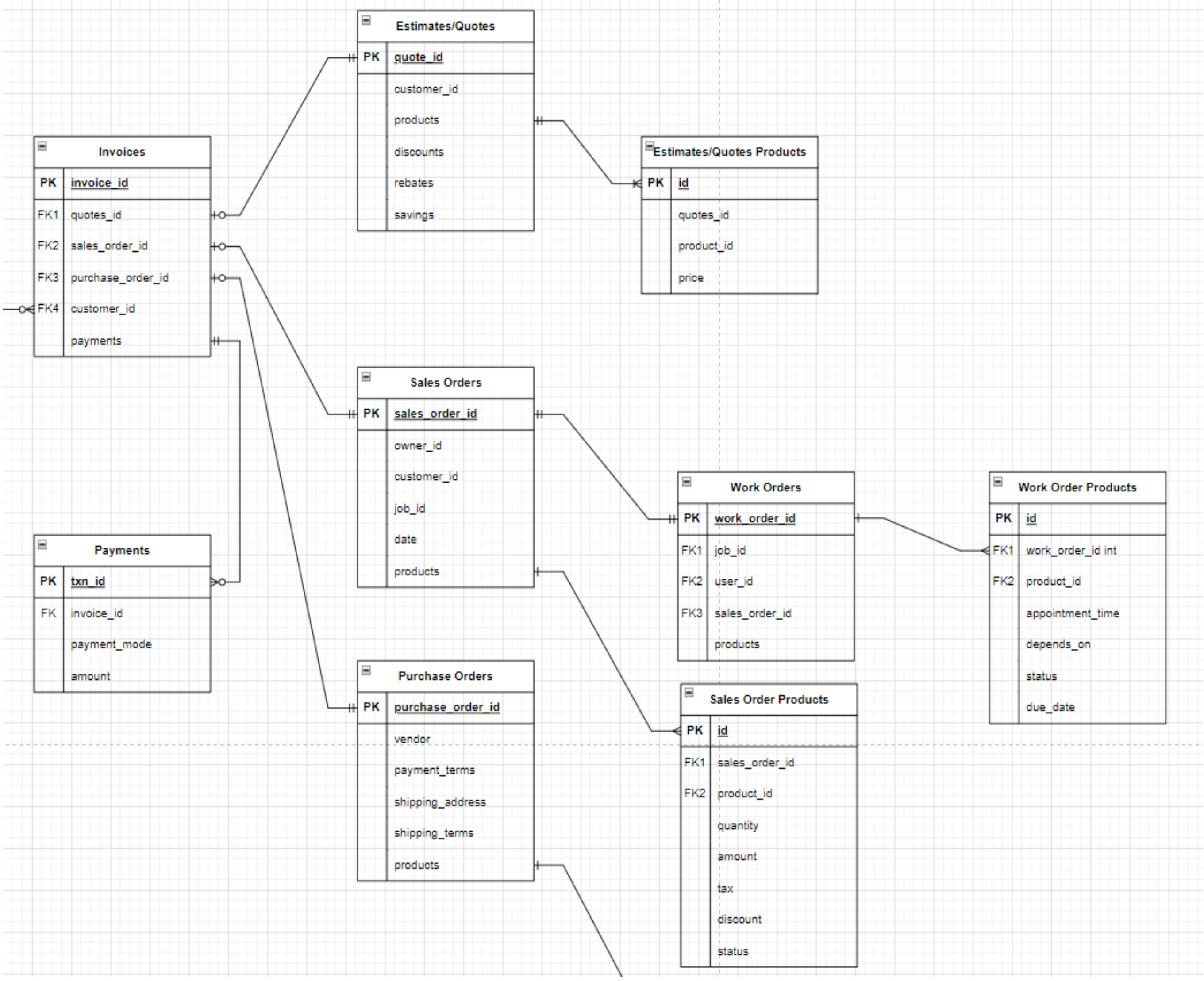


Figure 8. Orders, Invoices & Payments Data Model

The system will also include an Orders and Payments model, which is designed to manage and track sales, purchases, invoicing, and payments. This model includes several key components:

Estimates: Provides initial quotations to customers, giving them a preview of potential costs before finalizing a sale.

Sales Orders: Captures and manages sales information, detailing the specifics of customer orders.

Purchase Orders: Handles transactions with vendors, tracking purchases made for the business.

Work Orders: We introduce a new feature to support Field Service Management (FSM). Each Sales Order can be broken down into multiple Work Orders, with each Work Order representing a specific sub-job, such as installations or service calls.

All orders, whether sales or purchase, can be converted into invoices. The Payments module then tracks these invoices and records payments, ensuring comprehensive financial management within the system.

3.3 System Integration

Google

The integration includes several key Google services:

- **Google Meet**: This will allow the system to schedule and create virtual meetings with customers or internal teams directly within the platform. For instance, if a customer schedules an appointment or requires a consultation, a Google Meet link can be automatically generated and sent via email.
- **Google Slides**: Sales teams can create and share Google Slides presentations with customers for proposals, service explanations, or post-service reports. This makes it easy to present professional information in a structured format during or after customer interactions.
- **Google Maps**: Integrated into the system for tracking the addresses of customers and technicians, Google Maps provides real-time location data and routing information. This helps in dispatching technicians more efficiently by planning optimal routes and offering accurate directions to customers' locations.

SMS and Call

Twilio will be integrated with the system to make outbound calls and messages from within the CRM. As Twilio does not have an unlimited plan, and only pay as you go plan, the decision on whether to use Twilio or Ring central is still under discussion within the team.

Facebook Ads

Facebook ads will directly be integrated with the CRM, i.e. whenever a new lead is in the Facebook ads, the system will automatically create new records with Status New Lead. Unlike the old system, it will not require Zapier or Zoho Flow.

Adobe

Adobe will be used for the digital signing process, creating purchase agreements and sending it to the customers to sign.

Document Generation Process

Document Generation will be done via Agreements API. It accepts a document template (with template tags) along with the input data to generate the final document. The final document is generated by replacing all the template tags in the document template with dynamic content based on the actual values corresponding to the data input.

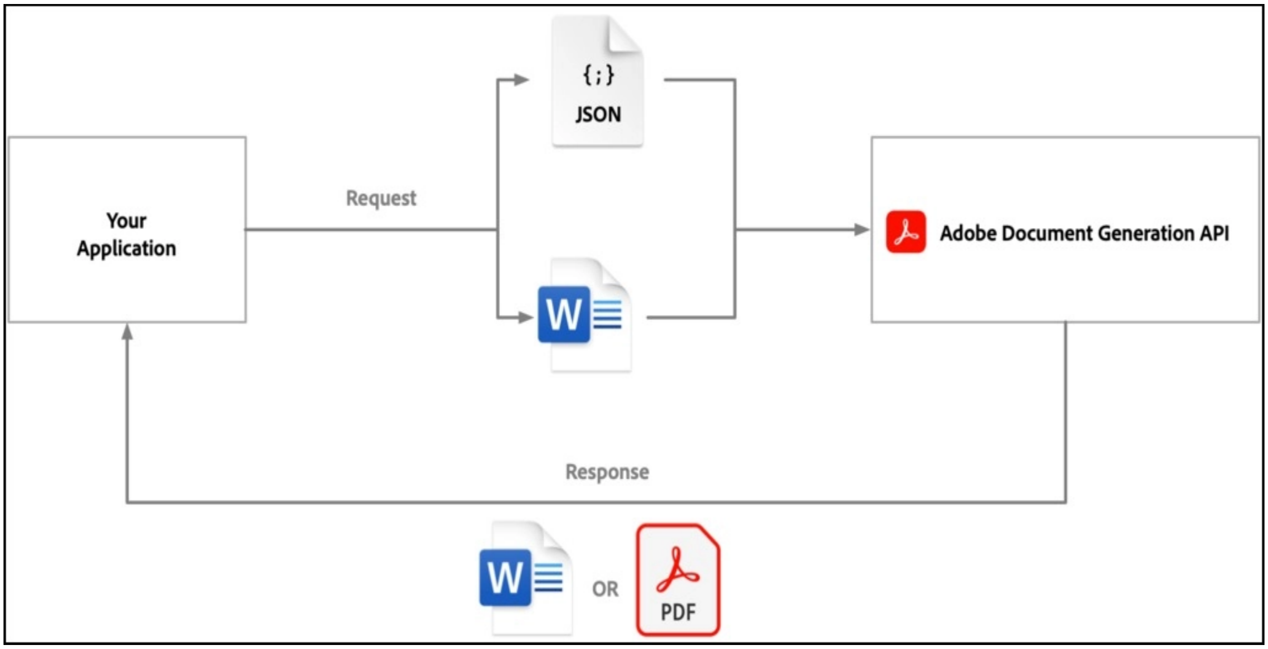


Figure 9. Document Generation Process

Adobe Sign also allows payment integration with their signed documents, which will be helpful for payment systems like security deposits while they sign the documents.

Payment

Square and Stripe will be used as payment gateways in order to charge the customers. As they are both popular for payments, both of the applications will be integrated to allow multiple payment options.

Orders API and Payments API with Square

The Square Orders API is used to create and manage orders, which represent customer purchases such as items, services, taxes, and discounts. To initiate a payment, an order is first created by sending a request to the Orders API with details such as the location ID and line items (items being purchased). This generates an `order_id` that links the order to any payment made. Once the order is created, the Square Payments API is used to process the payment by charging the customer. The payment request includes the `order_id` and a `source_id` (such as a card token) to associate the payment with the order.

After the payment is completed successfully, the order can be updated to reflect its final state (e.g., "COMPLETED"). This allows for proper tracking and management of orders within the system. You can retrieve the order details using the `order_id` to check the status or make updates, such as processing refunds. This flow ensures that each payment is tied directly to an order, providing a seamless experience for both merchants and customers, as well as accurate records for financial and inventory management.

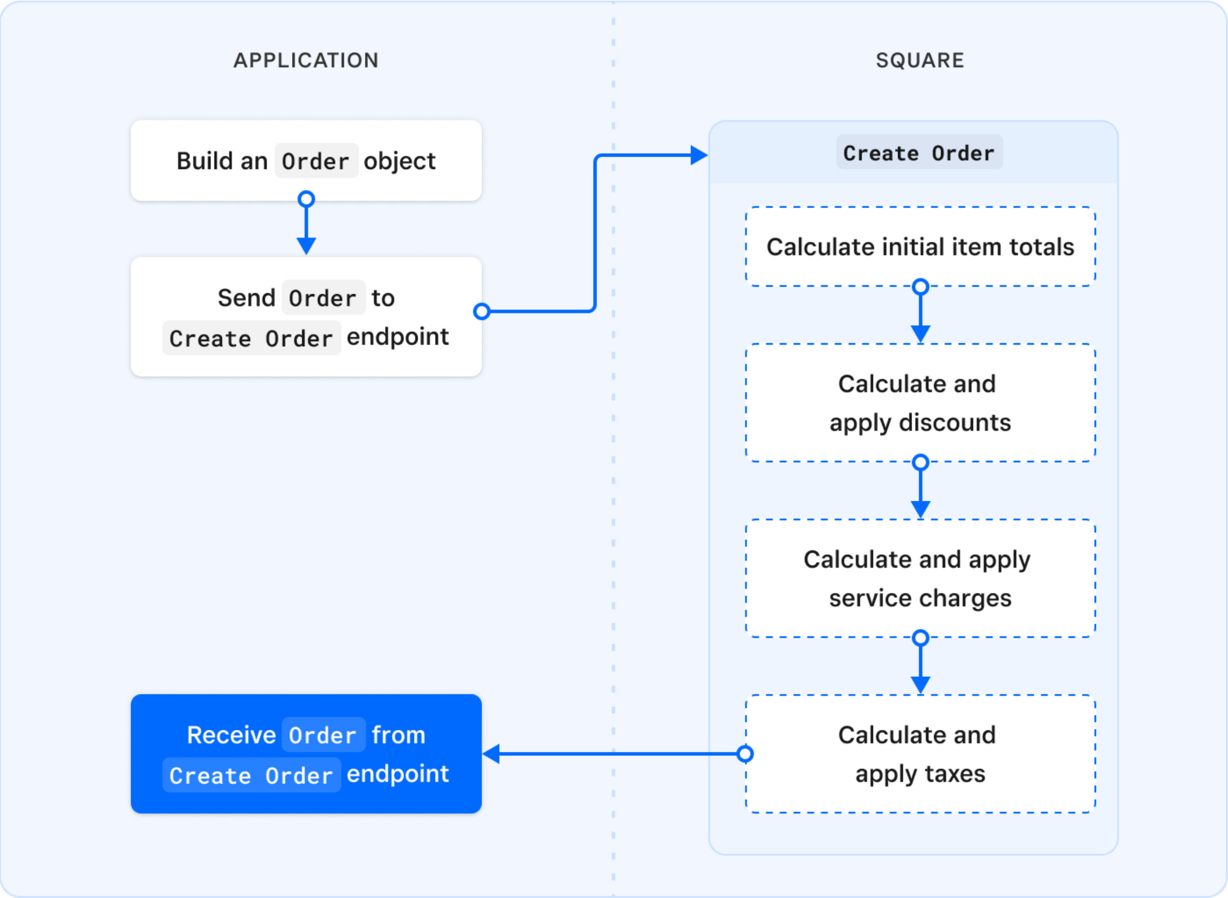

APPLICATION
SQUARE
Build an Order object
Send Order to Create Order endpoint
Create Order
Calculate initial item totals
Calculate and apply discounts
Calculate and apply service charges
Calculate and apply taxes
Receive Order from Create Order endpoint

Figure 10. Order API with Square

Stripe

The Stripe API can be seamlessly integrated into the system to facilitate payment processing. By utilizing the Stripe library, both server-side and client-side integration can be achieved. The payment form, created on the client side, captures payment details and sends them to the server as a token. On the server side, this token is used to create a charge through Stripe's API. Additionally, webhooks can be configured to handle various payment events and update the system accordingly. Testing in Stripe's test mode is recommended to ensure proper functionality before transitioning to live mode. With this integration, payments are processed securely and efficiently within the system.

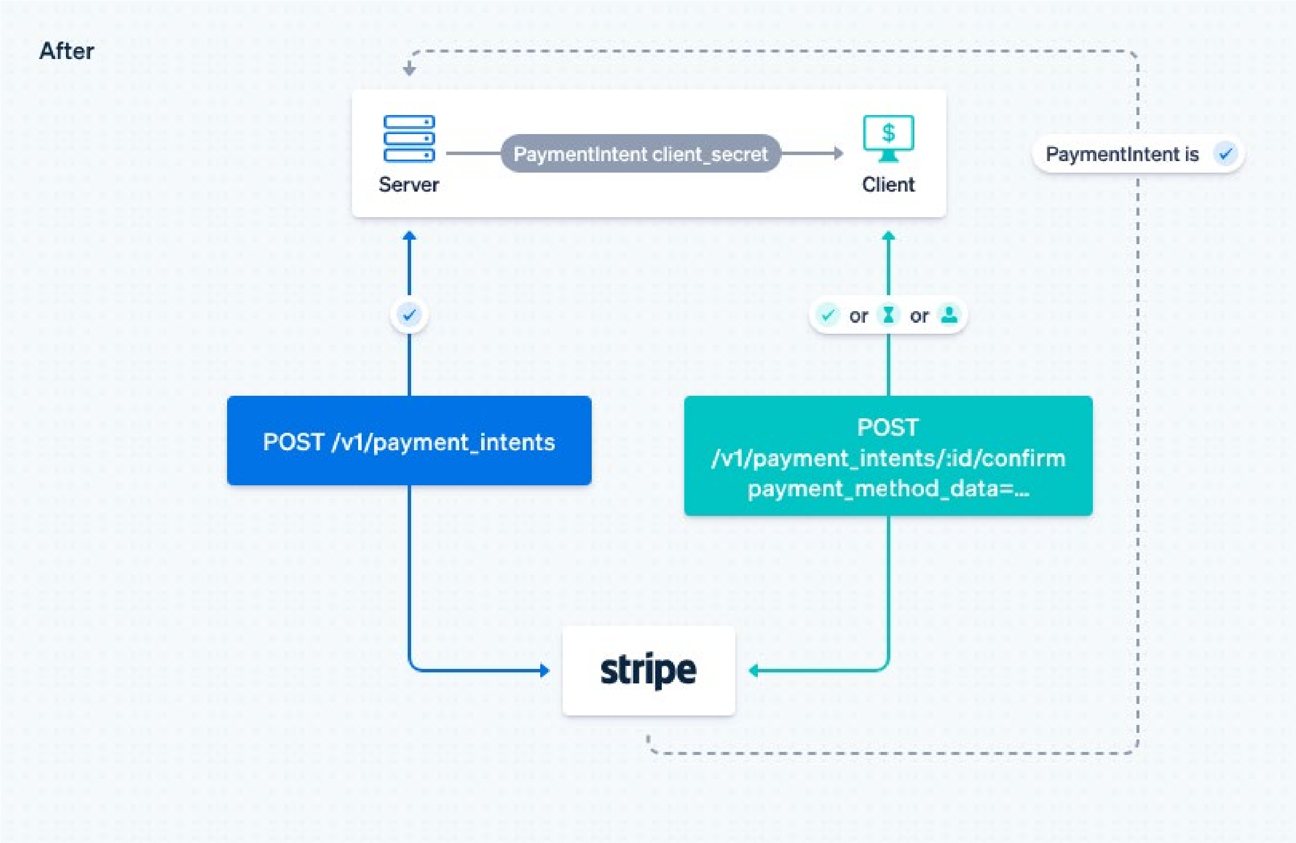


Figure 11. Stripe API

3.4 Data Workflow

3.4.1 Sales Pipeline

The well-established sales pipeline typically consists of the following stages: Lead Capture, Lead Qualification, Initial Contact, Appointment Booking, Needs Assessment, Proposal Generation, Follow-Up and Negotiation, and finally, Deal Closing. Through extensive research and industry analysis, several CRM platforms have refined this traditional model into a simplified four-stage process: Prospect Chosen, Meeting Agreed, Proposal Sent and Won.

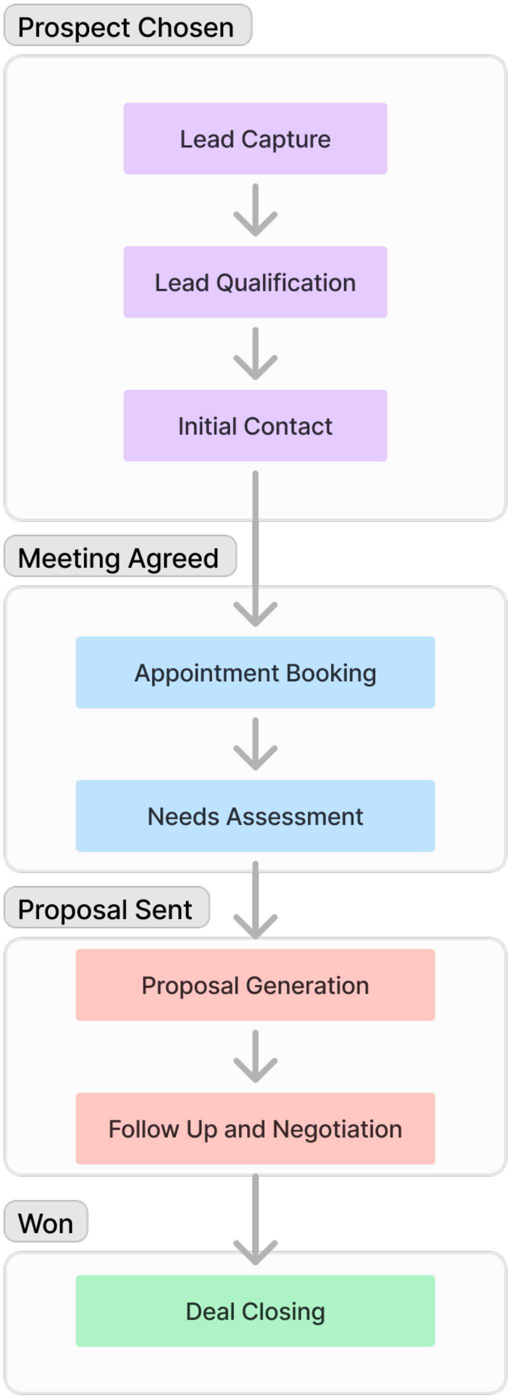


Figure 12. Conventional Sales Pipeline

In this streamlined version, Prospect Chosen consolidates the initial stages, encompassing both Lead Capture and Lead Qualification. The Meeting Agreed stage includes the scheduling of appointments, representing the formalization of the initial contact phase. Proposal Sent covers a broader range of activities, including Needs Assessment, Proposal Generation, and subsequent Follow-Up and Negotiation. The final stage, Won, mirrors the Deal Closing phase, signifying the successful conclusion of the sales process. This condensed approach retains the core components of the traditional pipeline while simplifying the flow, making it easier for CRM users to manage and track sales progress effectively.

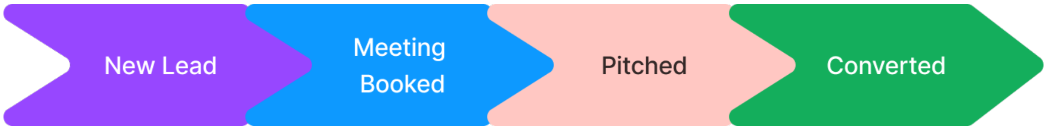


Figure 13. Simplified Sales Pipeline

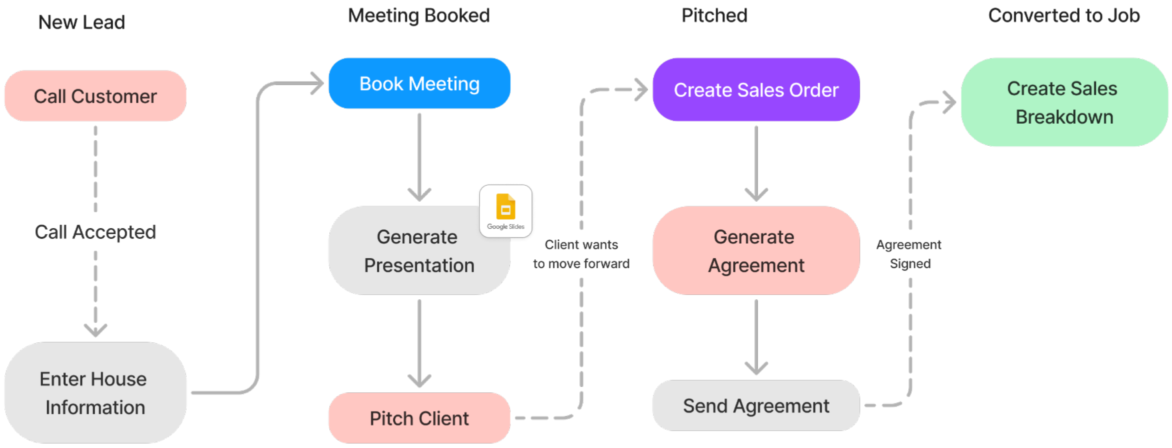


Figure 14. Sales Process

To have the naming convention simpler and more similar to the current CRM system in use, four methods will be used: New Lead, Meeting Booked, Pitched and Converted.

New Lead: This status will indicate all the new leads that come into the CRM system. These leads, similar to the current system, will be distributed to the agents based on a round robin process. The appointment booking, creating google meet, depending on if the meeting is virtual or not happens at this stage.

Meeting Booked: This status will indicate that the clients have an appointment with the sales agent, and the agents will be preparing a presentation, or pitch with the clients. If the meeting is virtual, the calculation, presentation generation, will all happen in this stage.

Pitched: This indicates that the agents have pitched the client and are waiting for the client to respond. All the follow ups will happen after this stage. The agents are required to create a sales order at this stage.

Converted: Converted is when the client is willing to move forward with the sales process. In order to convert the job, the agents, virtual or outside, will be required to upload a sales agreement document in the system, and enter the total sales amount.

3.4.2 Operation Pipeline

New Job

In this initial stage, a job request is created when a customer places an order for a service or installation. The job includes details such as customer information, service requirements, and location. This stage focuses on gathering all necessary information to proceed with planning and scheduling. The goal is to ensure the job is accurately logged into the system for further processing.

Create Work Order

Once the job is confirmed, a work order is generated. This work order serves as a comprehensive task list for the field technicians, outlining the scope of work, required materials, tools, and any special instructions. The work order also includes timelines and deadlines, ensuring the job proceeds according to schedule. This stage transforms the customer's request into actionable tasks.

Assign Technicians

In this phase, the work order is assigned to specific field technicians based on their skills, availability, and proximity to the job location. This assignment can be automated through a scheduling system that matches the job requirements with the best-suited technicians. Effective assignment ensures that the service is completed efficiently and meets the customer’s expectations.

Installed

This stage represents the execution phase, where the assigned technicians complete the installation or service at the customer's site. The technicians follow the instructions outlined in the work order, ensuring that the job is performed correctly, and any necessary materials are used. Progress is tracked in real-time, and any issues encountered are documented.

Closed

After the service is completed and confirmed to be successful, the job is marked as "Closed." The closed stage involves finalizing the work order, updating the job status, and processing any related paperwork, such as invoices or customer feedback. This stage ensures that the job is fully completed and all relevant parties are informed of the job's status.

3.5 Inventory Management

3.5.1 Stock In Process

When creating a purchase order, once the vendor has delivered the order, the system checks if the product item is configured to track serial numbers. If the "Track Serial Number" setting is enabled for the item, a prompt will pop up, requesting the user to scan the serial numbers of the delivered items. After scanning, new serial number records will be created for the items, and their status will be set to "INSTOCK" in the inventory system.

If the "Track Serial Number" setting is not enabled, the system will prompt the user to manually input the quantity of items delivered. Based on the information provided, the inventory quantity will be updated accordingly.

If the newly updated inventory quantity matches the ordered quantity, the order status will automatically change to "Delivered." However, if the delivered quantity is less than the ordered quantity, the status will be set to "Partially Delivered," indicating that not all items have been received yet.

This process ensures that both serialized and non-serialized items are accurately tracked and updated in the inventory system, reflecting their real-time stock levels after delivery.

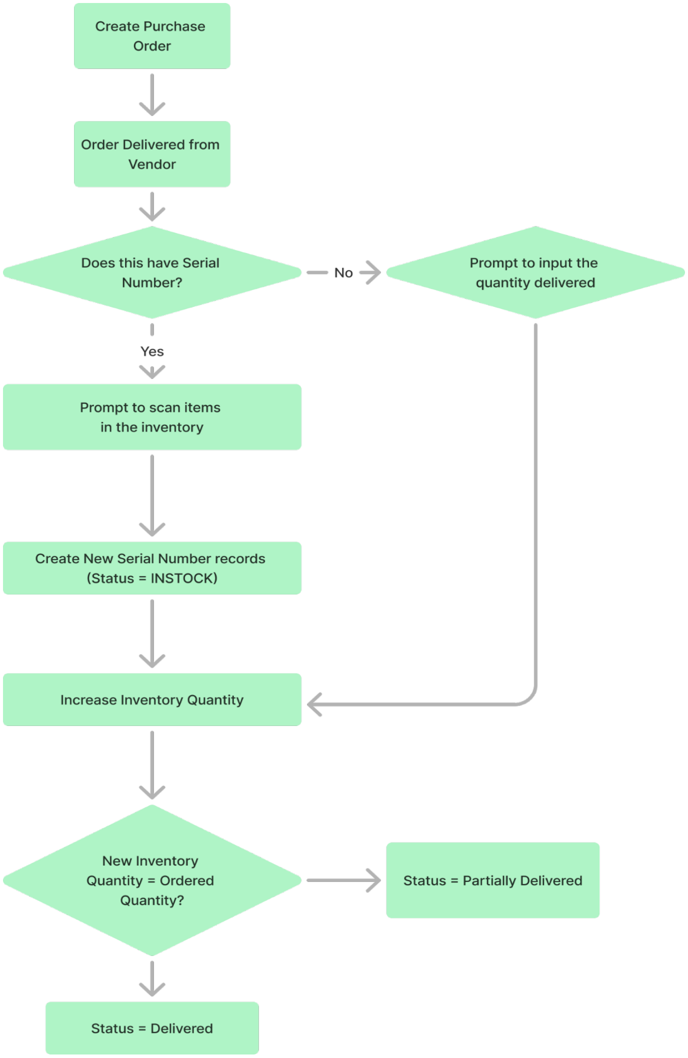

Create Purchase Order
Order Delivered from Vendor
Does this have Serial Number?
No
Prompt to input the quantity delivered
Yes
Prompt to scan items in the inventory
Create New Serial Number records (Status = INSTOCK)
Increase Inventory Quantity
New Inventory Quantity = Ordered Quantity?
Status = Partially Delivered
Status = Delivered

Figure 15. Stock in Process

3.5.2 Stock Out Process

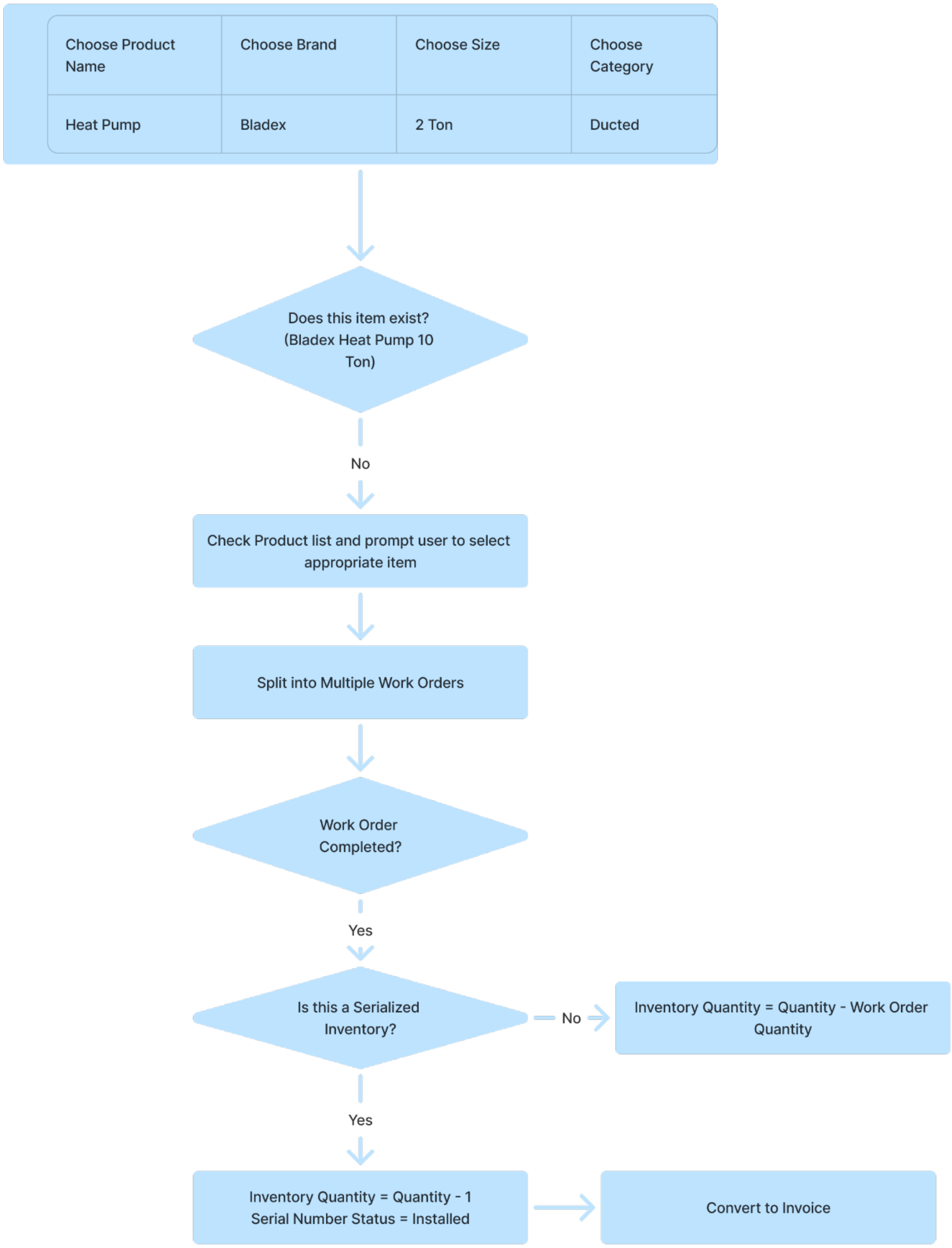
When selecting a product, the system first prompts the user to choose key attributes such as Product Name, Brand, Size, and Category. For example, the user may select "Heat Pump" as the product name, "Bladex" as the brand, "2 Ton" as the size, and "Ducted" as the category.

If the user then attempts to choose an item that does not exist in the current product list, such as "Bladex Heat Pump 10 Ton," the system will detect that the item is not available. It will then prompt the user to review the product list and select an appropriate item from the available options.

Once an appropriate item is selected, the system checks if the task requires splitting into multiple work orders. If so, it will manage and track the work orders accordingly.

When a work order is completed, the system further checks if the product is part of a serialized inventory. If it is not serialized, the inventory quantity is updated by subtracting the quantity used in the work order from the current stock. However, if the item is part of a serialized inventory, the system updates the inventory by reducing it by one unit for each completed work order, and it marks the corresponding serial number status as "Installed."

Finally, after completing the work order and adjusting the inventory, the system converts the work order into an invoice, ensuring the process from product selection to invoicing is efficiently managed.



Choose Product Name	Choose Brand	Choose Size	Choose Category
Heat Pump	Bladex	2 Ton	Ducted

Does this item exist? (Bladex Heat Pump 10 Ton)
No
Check Product list and prompt user to select appropriate item
Split into Multiple Work Orders
Work Order Completed?
Yes
Is this a Serialized Inventory?
No
Inventory Quantity = Quantity - Work Order Quantity
Yes
Inventory Quantity = Quantity - 1
Serial Number Status = Installed
Convert to Invoice

Figure 16. Stock Out Process

4. Languages, Databases and Frameworks

4.1 Primary Programming Languages

Python: We will use Python for its versatility and ease of integration with databases, APIs, and machine learning models. Its rich ecosystem of libraries and frameworks like Django, Fast API and Flask also allows for efficient back-end development and scalable CRM solutions.

JavaScript: JavaScript will be used for front-end development, providing dynamic and interactive user interfaces in the CRM. With frameworks like React and Node.js, it can also be utilized for full-stack development, ensuring a seamless user experience across web applications.

4.2 Backend and Frontend Frameworks

Django: Django is a powerful web framework that provides robust features like built-in authentication, database management, and security, making it ideal for developing the backend of our CRM. It also allows for the rapid creation of RESTful APIs, ensuring efficient communication between the front-end and back-end systems.

Next.js: Next.js is a React-based framework that offers server-side rendering (SSR) and static site generation (SSG), resulting in faster load times and improved SEO for the CRM. Its built-in routing, optimized performance, and ease of integration with APIs make it an excellent choice for building responsive and dynamic front-end applications.

4.3 Relational & No-SQL Databases

After careful research and analysis, we have concluded that a relational SQL database is the best fit for our CRM system, as data integrity is a critical factor. Therefore, for this project, we will proceed with PostgreSQL as a major database, a highly popular and reliable choice in the industry. Additionally, we will utilize NoSQL databases like Redis for caching, session management, and real-time data processing to enhance performance and scalability in the next phase of our development.

4.4 Cloud Services for Production

Azure App Services: Azure App Services will be used to host and deploy our CRM application on the cloud, ensuring scalable and reliable access to the system over the internet. It provides auto-scaling, load balancing, and security features to handle varying traffic levels effectively.

Azure Blob Storage: Azure Blob Storage will store large amounts of unstructured data, such as user documents, images, and backups. It offers secure and cost-effective storage, ensuring that the CRM can manage files efficiently and retrieve them as needed.

Azure Database (PostgreSQL): We will use Azure Database for PostgreSQL to manage and store the CRM's relational data. It provides high availability, security, and automated backups, ensuring that our customer and sales data is safely stored and can be accessed quickly when needed.

Azure Cache for Redis (No-SQL): We will use Azure Cache for Redis for caching, session control, and real-time processing in production. Redis is an open-source database service that helps to improve system performance by reducing database load, ensuring faster response times, and efficiently managing sessions and live data processing for the CRM.

5. Data Migration

Here is the plan for migrating the old CRM data into new ones:

5.1 Field Mapping: Old Field Names to New Field Names

To migrate data from the old CRM schema to the new one, a dictionary or lookup table will be created where old field names serve as keys, and the new field names act as corresponding values. This mapping dictionary will be utilized in the migration script to rename fields accordingly. For instance, the mapping might look like this:

```
field_mapping = {
    'old_field_1': 'new_field_1',
    'old_field_2': 'new_field_2',
}
new_data = old_data.rename(columns=field_mapping)
```

5.2 Blueprint Mapping: Old Blueprint to New Blueprint

This task involves mapping business processes or workflows from the old CRM to the new CRM. Key stages or process flows in both systems will be identified and compared. For each stage in the old CRM blueprint, it will be mapped to the corresponding stage or process in the new CRM. Additionally, any automations or triggers associated with these stages will be translated appropriately to ensure continuity in workflows.

5.3 Old Status Names → New Status Names

In the migration script, the old status names in `old_data` will be replaced with the new status names, and the updated data will be stored in `new_data`:

```
status_mapping = {
    'Old_Status_1': 'New_Status_1',
    'Old_Status_2': 'New_Status_2'
}
new_data['status'] = old_data['status'].replace(status_mapping)
```

5.4 Standardize Address → Address Module

Before importing addresses into the new CRM, they must be standardized. This involves breaking down address components—such as street, city, state, and zip code—into individual fields if required. Address verification services like Google Maps API or similar tools will be used to standardize and validate addresses. The addresses will be formatted according to the new CRM schema, and a new address record will be created:

```
new_data[['street', 'city', 'state', 'zip']] =
old_data['address'].str.split(',', expand=True)
```

5.5 Standardize Phone Numbers

Phone numbers will be normalized to a consistent format, such as the E.164 international format. The `phonenumbers` library in Python will be employed to validate and reformat phone numbers. Special characters, spaces, and inconsistencies will be removed. The following code snippet demonstrates this process:

```
import phonenumbers

def format_phone(number):
    parsed_number = phonenumbers.parse(number, "US")  # Use correct
country code
    return phonenumbers.format_number(parsed_number,
phonenumbers.PhoneNumberFormat.E164)

new_data['phone'] = old_data['phone'].apply(format_phone)
```

5.6 Tags for Imported Data

To differentiate data imported from the old CRM, a tag like "Imported from Old CRM" will be added to relevant records. This tagging helps in identifying the data source and managing it effectively within the new CRM.

5.7 Owner/Agent Mapping Based on Email Address

Data ownership or agent assignments will be mapped based on email addresses. A lookup table will be created to correlate old CRM email addresses with corresponding agents or owners in the new CRM. Records will be updated based on this mapping:

```
owner_mapping = {
    'old_email_1@example.com': 'new_owner_1',
    'old_email_2@example.com': 'new_owner_2'
}
new_data['owner'] = old_data['agent_email'].replace(owner_mapping)
```

5.8 Convert All Currency Data (Price, Cost, Amount) to Float or Currency Data Type

Price-related fields, such as price, cost, and amount, will be converted to a float or currency data type. Relevant fields will be identified, and any currency symbols or commas will be stripped out before conversion. The following code snippet shows how to convert price fields:

```
new_data['prhttps://docs.google.com/document/d/13j5WtzCFb4ldx2ywUo_Y6HEQYEMfj9ZPCUWYDI-9Xnc/editice'] = old_data['price'].replace({'\$': '', ',': ''}, regex=True).astype(float)
```

5.9 Standardize Date and Time Format

Date and time fields will be standardized to the format required by the new CRM, such as ISO 8601 or YYYY-MM-DD. Python’s `datetime` library or pandas will be used to convert and format these fields:

```
new_data['date'] = pd.to_datetime(old_data['date']).dt.strftime('%Y-%m-%d')
```

6. Cost Estimation and Project Timeline

6.1 Services Currently in Use

Table 3 Current Service Pricing

Service	Cost (Per Month)
RingCentral	CAD 690.35
Zapier	CAD 260 - CAD 270
Adobe Acrobat	CAD 40
GSuite	CAD 500
Zoho One	CAD 1200
KickServe	CAD 50
QuickBooks	CAD 170
Total	**CAD 2910.35 – CAD 3,000.35**

6.2 Services to Use

6.2.1 Azure Services Breakdown

This breakdown for azure is assumed to accommodate 50 users in the system.

Table 4 Azure Services Pricing

Service	Tier	Storage/Capacity	Cost per Unit	Estimated Cost	Estimated Cost in CAD
Azure App Services	Standard S1	For 50 users	$73/month	$73/month	CAD 98.55
Azure Blob Storage	Hot Access	500 GB	$0.0184 per GB/month	$9.20/month	CAD 12.42
Azure Database for PostgreSQL	General Purpose	250 GB	$147.68/month	$147.68/month	CAD 200.78
Azure Cache for Redis	Basic C0	250 MB	$16.43/month	$16.43/month	CAD 22.1805
Total Estimated Cost					CAD 333.64

Table 5 New Services Pricing

Service	Cost (Per Month)
Azure	CAD 300 - 400
Twilio	CAD 200
Adobe Acrobat	CAD 20
GSuite	CAD 500
Total Estimated	**CAD 520 + CAD 500 = 1020**

- **Current Total Monthly Cost: CAD 2910.35 – CAD 3,000.35**
- **Proposed Total Monthly Cost: CAD 1020 – CAD 1050**
- **Potential Savings:** CAD 1,990 per month

The proposed changes aim to cut down monthly operational costs by approximately **42-52%**, depending on the usage of Azure.

Additional Considerations

- **Zoho Subscription Growth:** As the company scales and adds more users, the cost of Zoho One will increase significantly due to its per-user pricing structure. A custom CRM, on the other hand, will offer more predictable and scalable pricing.
- **Scalability of Azure:** The cost for Azure may fluctuate between $300 and $400 per month based on usage, offering flexibility to scale resources as needed without the unpredictable cost increases tied to user numbers, as seen with Zoho.

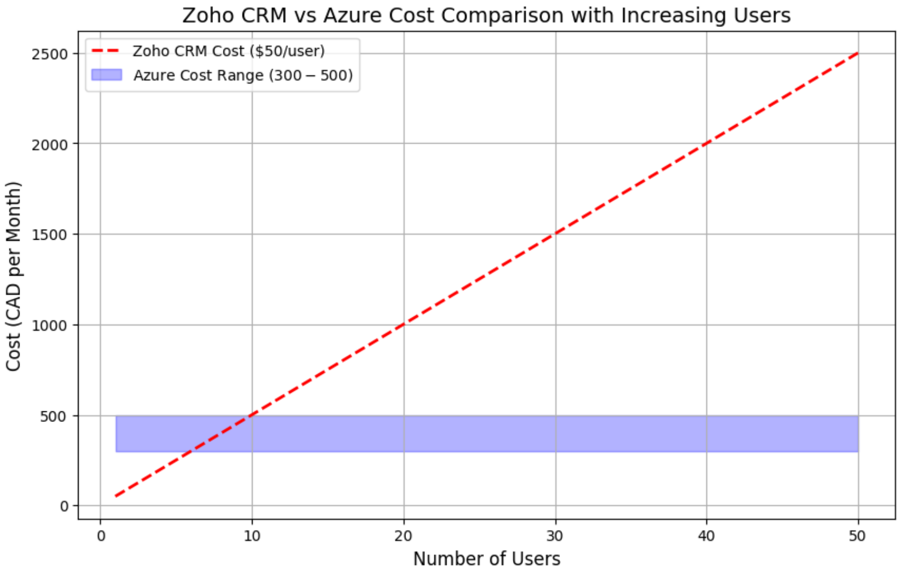


Figure 17 Zoho CRM vs Azure Cost Comaprison

6.3 Timeline

Project Plan and Milestones

Table 6 Project Plans and Milestones

Task	Start Date	End Date
Core Modules Development and Testing	2024-10-01	2024-12-15
Customer Management	2024-10-01	2024-11-01
Sales Order Processing	2024-11-02	2024-11-15
Work Order Generation	2024-11-16	2024-11-30
Technician Assignment	2024-12-01	2024-12-07
Product Inventory Management	2024-12-08	2024-12-15
Facebook Integration	2024-12-01	2024-12-07
Google Calendar Integration	2024-12-08	2024-12-10
Square Integration	2024-12-11	2024-12-15
Local Web CRM App Development	2024-11-01	2024-12-15
Email and SMS Communication	2024-12-01	2024-12-10
Outbound Calls	2024-12-11	2024-12-15
Backend Development	2024-10-01	2024-12-15
Frontend Development	2024-10-01	2024-12-15
Wireframing	2024-10-01	2024-10-15
Database Design	2024-10-16	2024-11-15
Cloud Services Integration	2025-01-01	2025-03-15
Migration to Cloud Environment	2025-01-01	2025-02-15
Staging Server Setup	2025-02-16	2025-03-01
Internal QA	2025-03-02	2025-03-15
Additional Mandatory Integrations	2025-02-16	2025-03-01
Additional Mandatory Features	2025-03-02	2025-03-15
Deployment in Production	2025-03-16	2025-03-31
Move to Live Production Environment	2025-03-16	2025-03-31
User Acceptance Testing (UAT)	2025-04-01	2025-04-10
User Feedback and Improvements	2025-04-11	2025-04-15

6.4 Support and Maintenance Plan

After the system is launched, consistent support and maintenance will be provided to the system in order to make sure the system is up and running. Following the deployment of the system, a comprehensive support and maintenance framework will be established to ensure optimal performance and reliability. The plan includes routine monitoring, anomaly detection, and system health checks to address potential issues proactively. The key components of the support and maintenance plan are outlined below:

6.5 Anomaly Detection and Alerts:

Real-time monitoring will be implemented to track the volume of leads created into the system within the defined time intervals.Automated Alerts will be triggered if the number of leads fall below the expected threshold average.

Automation and Functionality Monitoring: Monitoring mechanisms will be established in order to ensure that all the predefined automations and system functionalities are triggered as expected.If these automations malfunctions, system will trigger alerts

Function Failure Detection: All function errors are logged and the failures will trigger alerts and notifications.

API Request Monitoring: Monitor API fetch requests to ensure they complete within acceptable response time thresholds. Alerts will be set to trigger if any API requests exceed predefined latency limits, indicating potential performance bottlenecks or connectivity issues.

Status Duration Monitoring: Track records and their status transitions within the system. Alerts will be generated if records remain in a particular status for an extended period, which may indicate potential process inefficiencies or data entry issues.

6.6 User Support and Continuous Improvement:

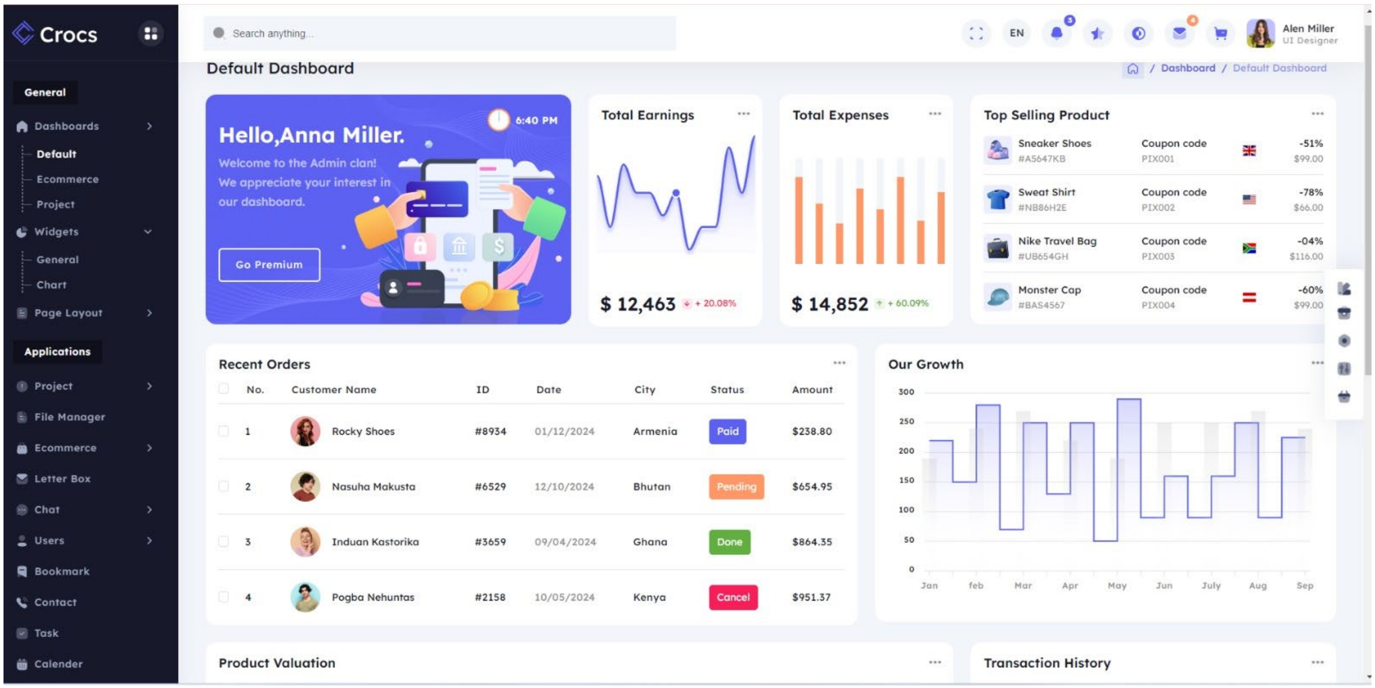
Proper documentation and Training: Training for all the user groups will be provided and each step will be well documented in terms of the user point of view and also the developer’s point of view. All the video tutorials will be stored in the training drive.

Feedback and Reviews: The IT department will make sure to take the feedback from users and identify areas for development. Enhancements based on performance metrics along with user experience will be used to continuously improve the system.

7. User Experience and Design Considerations

Following Considerations are made to accommodate User Experience.

- Improve page navigation (Minimize number of clicks)
- Faster page/data reloading / real-time updates
- Minimal UI, make only the relevant modules visible in the user screen
- Support shortcut hotkeys to improve efficiency
- Advanced table search/filtering/sorting/pagination/pinning
- Search bar to quickly navigate between screens
- Proper font selection to improve readability
- Responsive design
- Automated workflow tasks
- Detailed activity logs
- Use of distinct icons/colors
- Integration of 3rd party tools
- Kanban views



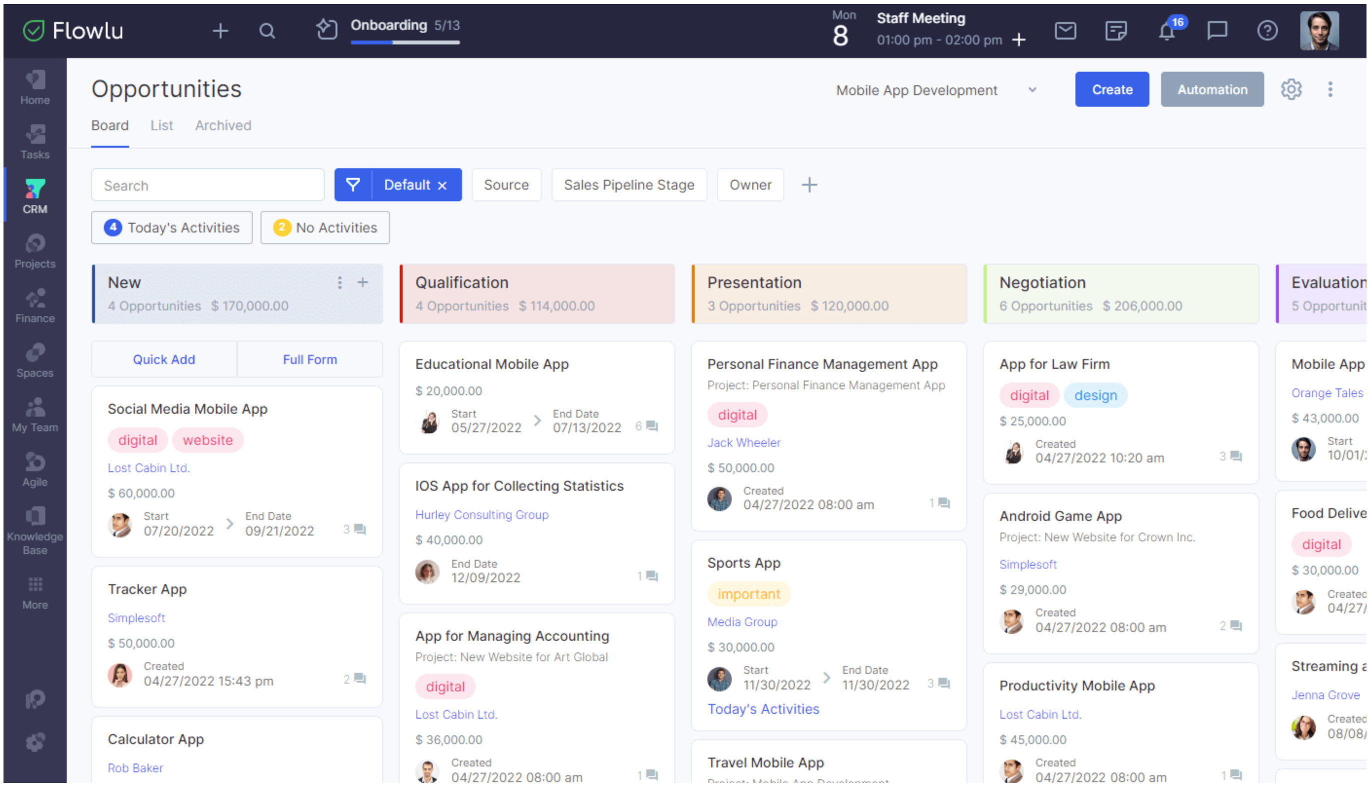
Top Selling Product

Sneaker Shoes #A5647KB	Coupon code PIX001	🇬🇧	-51% $99.00
Sweat Shirt #NB86H2E	Coupon code PIX002	🇺🇸	-78% $66.00
Nike Travel Bag #UB654GH	Coupon code PIX003	🇿🇦	-04% $116.00
Monster Cap #BAS4567	Coupon code PIX004		-60% $99.00

Recent Orders

□	No.	Customer Name	ID	Date	City	Status	Amount
□	1	Rocky Shoes	#8934	01/12/2024	Armenia	Paid	$238.80
□	2	Nasuha Makusta	#6529	12/10/2024	Bhutan	Pending	$654.95
□	3	Induan Kastorika	#3659	09/04/2024	Ghana	Done	$864.35
□	4	Pogba Nehuntas	#2158	10/05/2024	Kenya	Cancel	$951.37

Figure 18 Targeted UI/UX Design1



New
4 Opportunities $ 170,000.00

Social Media Mobile App
digital website
Lost Cabin Ltd.
$ 60,000.00
Start 07/20/2022 > End Date 09/21/2022 3

Tracker App
Simplesoft
$ 50,000.00
Created 04/27/2022 15:43 pm 2

Calculator App
Rob Baker

Qualification
4 Opportunities $ 114,000.00

Educational Mobile App
$ 20,000.00
Start 05/27/2022 > End Date 07/13/2022 6

IOS App for Collecting Statistics
Hurley Consulting Group
$ 40,000.00
End Date 12/09/2022 1

App for Managing Accounting
Project: New Website for Art Global
digital
Lost Cabin Ltd.
$ 36,000.00
Created 04/27/2022 08:00 am 1

Presentation
3 Opportunities $ 120,000.00

Personal Finance Management App
Project: Personal Finance Management App
digital
Jack Wheeler
$ 50,000.00
Created 04/27/2022 08:00 am 1

Sports App
important
Media Group
$ 30,000.00
Start 11/30/2022 > End Date 11/30/2022 3
Today's Activities

Travel Mobile App

Negotiation
6 Opportunities $ 206,000.00

App for Law Firm
digital design
$ 25,000.00
Created 04/27/2022 10:20 am 3

Android Game App
Project: New Website for Crown Inc.
Simplesoft
$ 29,000.00
Created 04/27/2022 08:00 am 2

Productivity Mobile App
Lost Cabin Ltd.
$ 45,000.00
Created 04/27/2022 08:00 am 1

Evaluation
5 Opportunit

Mobile App
Orange Tales
$ 43,000.00
Start 10/01/

Food Delive
digital
$ 30,000.00
Created 04/27/

Streaming a
Jenna Grove
Created 08/08/

Figure 19 Targeted UI/UX Design2

8. CRM Development Challenges

Figure 20 CRM Development Challenges

1. Data Migration: Transferring data from the old system to the new CRM can be complex, especially when dealing with large data sets. Ensuring data integrity, accuracy, and compatibility during the migration is a major challenge.

2. Integration with Existing Systems: The new CRM must seamlessly integrate with other tools, such as marketing, sales, and customer support platforms, which can be difficult depending on the legacy system's architecture.

3. User Adoption and Training: Shifting users from the old system to the new CRM requires proper training and change management to ensure smooth adoption and minimize disruption to business processes.

4. Robust Pipeline Design: Designing and implementing a robust pipeline for handling customer data, sales processes, and automation features requires careful planning to avoid performance bottlenecks and ensure scalability.

5. Time Constraints: Managing the project within the allocated time frame (3-4 months for 1st phase development and 7-8 months for full phase development with a team of 3-4 members) while balancing development, testing, and iterations to meet deadlines and deliver a functional CRM system.

9. Future Work for CRM Development

1. Import and Exporting Data in Different File Formats: We will enhance the CRM's data management capabilities by supporting a wide range of file formats beyond CSV, such as Excel, XML, and JSON. Additionally, integrating an AI system to scan and extract data from images will facilitate the automatic import of data from scanned documents and photos, streamlining the data entry process.

2. Integrating AI Voice Calling System: To improve customer support and appointment scheduling, we will implement an AI-driven voice calling system. This system will handle automated calls, provide customer support, and schedule appointments, leveraging natural language processing to interact with users effectively.

3. Automated AI-Generated Emails and SMS Templates: We plan to incorporate AI-generated templates for emails and SMS communications within the CRM. This will allow for automated and personalized messaging, enhancing communication efficiency and consistency while reducing manual effort.

4. Integrating Notification Triggers for Critical Scenarios: The CRM will feature notification triggers based on anomaly detection to alert users of critical scenarios. By analyzing data patterns and identifying deviations, the system will provide real-time alerts for potential issues, helping to proactively address problems.

5. AI-Based Documentation System: We will develop an AI-powered documentation system to guide CRM users. This system will offer contextual help and automated documentation, making it easier for users to navigate and utilize the CRM's features effectively.

6. Leveraging Open Source for Cost Reduction: To reduce costs, we will aim to use open-source technologies and tools wherever possible. This approach will help in minimizing licensing fees and promote flexibility in customizing the CRM to meet specific needs.

7. Developing Apps for Technicians for FSM: We will create mobile applications specifically for technicians involved in Field Service Management (FSM). These apps will facilitate job management, real-time updates, and efficient communication, enhancing the overall efficiency of field operations.

8. Developing a Client-Oriented App for Online and Offline Selling: A client-oriented app will be developed to support both online and offline selling. This app will provide customers with a seamless shopping experience, including features for browsing products, making purchases, and accessing order information both online and offline.

9. Tracking CRM Users' Locations in Real Time: We will integrate real-time location tracking using Google Maps to monitor CRM users' locations. This feature will support logistical planning, field operations management, and ensure better coordination and efficiency in managing customer interactions and service delivery.

References

Murphy, M. (2020). How to develop a CRM application: A step-by-step guide. Developer's Journal. https://www.developersjournal.com/how-to-develop-crm-application

Smith, J. (2019). Best practices for CRM software development. Tech Innovations. https://www.techinnovations.com/best-practices-crm-software-development

Williams, K. (2021). Choosing the right CRM development tools. CodeCraft. https://www.codecraft.com/choosing-right-crm-tools

Lee, A. (2022). Understanding CRM software architecture. Software Engineering Today. https://www.softwareengineeringtoday.com/crm-software-architecture

Garcia, R. (2021). Integration of third-party services in CRM applications. Dev Insights. https://www.devinsights.com/third-party-crm-integration

Patel, S. (2020). Designing user-friendly interfaces for CRM systems. UX Developer Hub. https://www.uxdeveloperhub.com/user-friendly-crm-interfaces

Jones, L. (2019). The role of cloud services in CRM development. Cloud Computing Review. https://www.cloudcomputingreview.com/cloud-services-crm-development

Adams, T. (2022). Implementing robust security measures in CRM software. Security Dev Weekly. https://www.securitydevweekly.com/robust-crm-security

Chen, Y. (2021). Optimizing database performance for CRM applications. Database Developer Digest. https://www.databasedevdigest.com/optimizing-crm-database

Johnson, E. (2023). Leveraging APIs for CRM integrations. API Developer Journal. https://www.apideveloperjournal.com/crm-api-integrations