SAKSHAM APPLICATION

PROJECT

X`

Document 1: Definition of Done

In Agile, **DOD** stands for **Definition of Done**. It is a critical concept that defines the criteria that must be met for a task, user story, feature, or product increment to be considered complete. DOD helps the team ensure that work is finished to the required standard and that it is potentially shippable or ready for delivery.

1. Acceptance Criteria (To Satisfy Customer Requirements)

Acceptance Criteria (AC) are the conditions that must be met for a user story or product feature to be considered complete from the customer's or product owner’s perspective. These criteria are usually defined in collaboration between the Product Owner and the team, and they focus on the functional aspects of the product, ensuring that the customer’s needs are met.

**User Story:**  
“As a customer , I want to login into bank mobile application so that I can View my account details in my dashboard.

Acceptance Criteria:

The user must be able to sign up using Customer ID and password.

User ID must Be customer ID, and passwords (less than 8 characters including Alpha numeric)

Quality Criteria (To Satisfy Quality Requirements)

**Quality Criteria** focus on the **non-functional requirements** that ensure the product is built to a certain standard of **performance, reliability, security, usability, and maintainability**. These are the standards that help ensure the product is of high quality and can be effectively used and maintained in a production environment.

#### **Key Aspects of Quality Criteria:**

* **Non-Functional Requirements**: Quality criteria include aspects like system performance, load handling, security, and accessibility. They ensure that the product works well under various conditions, is secure, and provides a good user experience.
* **Performance and Reliability**: These criteria ensure that the product meets certain performance benchmarks, like fast loading times, handling a certain number of users concurrently, or maintaining uptime.
* **Testability and Usability**: Quality criteria also include how easy it is to use the product and whether it meets usability standards (user-friendly, intuitive design).
* **Security and Compliance**: Security features, such as data encryption, secure authentication, and privacy measures, are also part of the quality criteria.

Document 2- Product Vision

### **Product Vision :**

The **Saksham Application** aims to empower **Axis Bank employees** by providing a seamless, efficient, and user-friendly digital platform to enhance productivity, streamline internal processes, and support professional growth. It envisions an integrated system that facilitates employee engagement, performance tracking, learning & development, and operational efficiency.

With a focus on **automation, accessibility, and real-time insights**, Saksham will serve as a central hub for managing HR services, compliance tracking, and skill enhancement initiatives. The application will leverage **AI-driven analytics, secure data management, and intuitive UI/UX** to create a smooth experience for employees.

The long-term vision is to **enhance workforce efficiency, drive digital transformation, and foster a culture of continuous improvement**, ensuring that Axis Bank employees remain agile, informed, and empowered in an evolving banking landscape.

| Field | Details |
| --- | --- |
| Product Vision | Saksham Application – A platform to view the entire customer portfolio in details with linked mobile number. |
| Scrum Project | Saksham Application |
| Name | Saksham |
| Venue | Hyderabad, India |
| Date | 2025-01-20 |
| Start time | 10:00 AM |
| End time | 05:00 PM |
| Duration | 7 hours |
| Client | Saksham Technologies Pvt. Ltd. |
| Stakeholder list | Mr. Ramesh (Product Owner), Ms. Priya (Client Representative), Mr. Naveen (Marketing Head), Ms. Ayesha (UX Designer), Mr. Sai Kiran (Business Analyst), Mr. Vikram (Test Lead) |

| Scrum Team | Details |
| --- | --- |
| Scrum Master | Mr. Sandeep Kumar |
| Product Owner | Mr. Ramesh Kumar |
| Scrum Developer 1 | Mr. Arun Patel |
| Scrum Developer 2 | Ms. Neha Sharma |
| Scrum Developer 3 | Mr. Pradeep Reddy |
| Scrum Developer 4 | Ms. Aishwarya Rao |
| Scrum Developer 5 | Mr. Vijay Kumar |

Vision: What is your vision, your overarching goal for creating the product?

| Section | Details |
| --- | --- |
| Vision | The Saksham Application aims to streamline and simplify how Axis Bank employees access, view, and manage customer portfolios. By providing a centralized view of customer data, the app enhances productivity and supports employees in delivering timely and efficient customer service. |
| Target Group | The target users are Axis Bank employees, particularly relationship managers, account managers, and customer service representatives who need quick access to detailed customer data (e.g., current accounts, savings accounts) to offer personalized banking services. |
| Market Segment | The product addresses the banking and financial services market, focusing on enhancing internal operations and improving employee efficiency. It is specifically tailored to meet the needs of Axis Bank employees. |
| Target Users/Customers | Primary users: Axis Bank employees (relationship managers, customer service reps, account managers, etc.). |
| Needs | The Saksham application solves the problem of incomplete and delayed access to customer data. Bank employees often need to manually search for customer information across different systems, which can lead to delays and errors. This application provides an integrated, real-time view of the entire customer portfolio, enabling employees to quickly access relevant information and serve customers more effectively. |
| Benefit | The app provides a range of benefits, including:  1. Faster access to customer portfolios: Employees can instantly access current and savings account details.  2. Improved efficiency: Reduces the need for employees to toggle between multiple systems or databases.  3. Better customer service: Empowering employees with real-time data to deliver accurate and personalized solutions to customers. |
| Product | The Saksham Application is a Web based tool designed for Axis Bank employees to view customer portfolios. It integrates with Axis Bank's internal databases to show information about registered numbers, customer accounts, transaction history, and more, in a user-friendly interface. |
| What Makes it Special | The app's ability to seamlessly integrate with Axis Bank’s internal systems, providing real-time data and easy access to comprehensive customer profiles is what makes it special. The user-friendly interface and quick retrieval of data without the need for switching between multiple platforms ensure it stands out in terms of efficiency. |
| Feasibility | The development of the Saksham Application is feasible, as Axis Bank already has internal systems and databases that can be integrated. The application will be developed using standard technologies such as mobile app development frameworks (React Native, Flutter) and back-end APIs for database interaction. |
| Value to the Company | The Saksham Application will significantly enhance the efficiency of Axis Bank employees, leading to better customer service and higher customer satisfaction. By reducing the time spent on manual data retrieval, employees can focus on more value-added tasks, ultimately improving productivity and profitability for the bank. |
| Business Goals | - Improve employee productivity by providing quick access to comprehensive customer data.  - Enhance customer satisfaction through faster, more accurate service.  - Reduce operational inefficiencies and streamline internal processes. |
| Business Model | The business model for Saksham focuses on improving internal operational efficiency rather than generating direct revenue. The app will be free for internal use, but it indirectly benefits the business by improving employee efficiency, reducing service response times, and enhancing customer satisfaction. |

Document 3: User stories

**As a user**, I want to log in securely to the Saksham application, so that I can access my customer portfolio and other relevant information.

BV: 10 CP: 20

Acceptance criteria:

1. The login page must have fields for **username** (employee ID) and **password**.   
   2. The system should authenticate using Axis Bank's internal user authentication system.   
   3. If login credentials are valid, the user should be redirected to the **dashboard** where they can view the customer portfolio.   
   4. If login credentials are invalid, the system should display an **error message** such as "Invalid username or password."   
   5. The system should provide an option for users to **reset their password** via a secure recovery process.   
   6. The login page should have **security features** like encryption to protect sensitive employee data.   
   7. The login process should complete within **5 seconds** to ensure smooth user experience.

**User Story No:** US001 **Tasks:** View customer profile summary. **Priority:** High **Value Statement:** As a user, I want to view a customer's profile summary, so that I can quickly access essential customer details.

**BV:** 500 **CP:** 3

**Acceptance Criteria:**

1. The profile summary must display the customer's name, account number, and linked mobile number.
2. The interface should load within 2 seconds.
3. Data should be accurate and updated in real-time.

#### User Story 2

**User Story No:** US002 **Tasks:** Search customers by mobile number. **Priority:** High **Value Statement:** As a user, I want to search for a customer using their mobile number, so that I can locate specific customer profiles efficiently.

**BV:** 500 **CP:** 5

**Acceptance Criteria:**

1. The search bar should support mobile number input.
2. Results must be displayed instantly upon entering the mobile number.
3. Error messages should appear for invalid or unregistered mobile numbers.

#### User Story 3

**User Story No:** US003 **Tasks:** View transaction history. **Priority:** Medium **Value Statement:** As a user, I want to view the transaction history of a customer, so that I can analyze their financial activities.

**BV:** 200 **CP:** 4

**Acceptance Criteria:**

1. The transaction history must display date, type, amount, and status.
2. Filters should be available for date range and transaction type.
3. Data should be exportable in PDF and Excel formats.

#### User Story 4

**User Story No:** US004 **Tasks:** Add customer notes. **Priority:** Medium **Value Statement:** As a user, I want to add notes to a customer's profile, so that I can record additional details or follow-up actions.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Notes must support text input up to 500 characters.
2. Notes should display the timestamp and user who added them.
3. Notes must be editable and deletable by authorized users.

#### User Story 5

**User Story No:** US005 **Tasks:** Assign a relationship manager to a customer. **Priority:** High **Value Statement:** As a user, I want to assign a relationship manager to a customer, so that the customer receives personalized support.

**BV:** 1000 **CP:** 6

**Acceptance Criteria:**

1. A dropdown should display available relationship managers.
2. The system should send a notification to the assigned manager.
3. Assignment details must be visible in the customer's profile.

#### User Story 6

**User Story No:** US006 **Tasks:** Display customer portfolio value. **Priority:** High **Value Statement:** As a user, I want to see the overall portfolio value of a customer, so that I can evaluate their financial position.

**BV:** 100 **CP:** 2

**Acceptance Criteria:**

1. Portfolio value must aggregate across all accounts linked to the customer.
2. Data should refresh in real-time.
3. Values must display currency formatting.

#### User Story 7

**User Story No:** US007 **Tasks:** Notify customers of portfolio updates. **Priority:** Medium **Value Statement:** As a user, I want to send notifications to customers about changes in their portfolio, so that they stay informed about updates.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Notifications should be sent via email and SMS.
2. Messages must include details of the update and a link to the portfolio.
3. Notification logs must be accessible for compliance.

#### User Story 8

**User Story No:** US008 **Tasks:** Generate portfolio performance reports. **Priority:** Medium **Value Statement:** As a user, I want to generate detailed reports of a customer's portfolio performance, so that I can share insights with the customer.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Reports must include graphs and tables summarizing key metrics.
2. Reports should be downloadable in PDF format.
3. Users should have the option to email reports directly to customers.

#### User Story 9

**User Story No:** US009 **Tasks:** Display linked accounts. **Priority:** High **Value Statement:** As a user, I want to see all accounts linked to a customer, so that I can understand their relationship with the organization.

**BV:** 200 **CP:** 3

**Acceptance Criteria:**

1. Linked accounts must display account type, number, and balance.
2. Details must update in real-time.
3. A count of total linked accounts must be displayed.

#### User Story 10

**User Story No:** US010 **Tasks:** Enable bulk customer uploads. **Priority:** Low **Value Statement:** As a user, I want to upload customer data in bulk, so that I can save time when managing multiple records.

**BV:** 500 **CP:** 3

**Acceptance Criteria:**

1. Uploads must support CSV and Excel formats.
2. Error logs must highlight issues in the uploaded data.
3. Successfully uploaded records must auto-link to existing profiles.

#### User Story 11

**User Story No:** US011 **Tasks:** Filter customer portfolio by investment type. **Priority:** Medium **Value Statement:** As a user, I want to filter a customer's portfolio by investment type, so that I can focus on specific financial instruments.

**BV:** 500 **CP:** 5

**Acceptance Criteria:**

1. Filters must include categories like equities, fixed deposits, and mutual funds.
2. The filter should apply instantly and update the display.
3. The system should allow multi-select filtering.

#### User Story 12

**User Story No:** US012 **Tasks:** Provide customer risk profiling. **Priority:** High **Value Statement:** As a user, I want to view the risk profile of a customer, so that I can recommend suitable financial products.

**BV:** 200 **CP:** 3

**Acceptance Criteria:**

1. The risk profile should be calculated based on customer data and investment patterns.
2. Risk levels must be categorized as low, medium, or high.
3. Recommendations must be generated based on the risk profile.

#### User Story 13

**User Story No:** US013 **Tasks:** Enable customer data validation. **Priority:** Medium **Value Statement:** As a user, I want to validate customer details before submission, so that I can ensure data accuracy.

**BV:** 500 **CP:** 3

**Acceptance Criteria:**

1. Validation must include checks for mandatory fields, formats, and duplicates.
2. Errors should be highlighted with detailed messages.
3. Submission should only be allowed after successful validation.

#### User Story 14

**User Story No:** US014 **Tasks:** Integrate third-party KYC verification. **Priority:** High **Value Statement:** As a user, I want to perform KYC verification via a third-party service, so that I can comply with regulatory requirements.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Integration must support APIs from leading KYC providers.
2. The system should display verification status in the customer profile.
3. Notifications must be sent for successful or failed KYC.

#### User Story 15

**User Story No:** US015 **Tasks:** Allow role-based access control. **Priority:** High **Value Statement:** As an admin, I want to manage role-based access for users, so that I can maintain data security.

**BV:** 500 **CP:** 3

**Acceptance Criteria:**

1. Access control must include predefined roles such as admin, manager, and viewer.
2. Permissions must be configurable by module.
3. Audit logs must track access and changes.

#### User Story 16

**User Story No:** US016 **Tasks:** Display customer demographic information. **Priority:** Medium **Value Statement:** As a user, I want to view demographic details of a customer, so that I can tailor communications effectively.

**BV:** 500 **CP:** 5

**Acceptance Criteria:**

1. Demographic information must include age, location, and income range.
2. Details must be editable by authorized users.
3. Updates should be reflected across linked accounts.

#### User Story 17

**User Story No:** US017 **Tasks:** Add customer referrals. **Priority:** Low **Value Statement:** As a user, I want to record customer referrals, so that I can track referral-based growth.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Referrals must link to existing or new profiles.
2. Referral rewards must be tracked and displayed.
3. Notifications must be sent for successful referrals.

#### User Story 18

**User Story No:** US018 **Tasks:** Export customer data. **Priority:** Medium **Value Statement:** As a user, I want to export customer data in various formats, so that I can analyze it offline.

**BV:** 2000 **CP:** 13

**Acceptance Criteria:**

1. Supported formats must include CSV, Excel, and JSON.
2. Exported data must include all visible columns.
3. A download history log must be maintained.

#### User Story 19

**User Story No:** US019 **Tasks:** Monitor customer activity logs. **Priority:** High **Value Statement:** As a user, I want to monitor activity logs for customer interactions, so that I can track actions for compliance.

**BV:** 500 **CP:** 5

**Acceptance Criteria:**

1. Logs must include timestamps, actions, and user IDs.
2. Filters must be available for date range and action type.
3. Logs must be exportable for audits.

#### User Story 20

**User Story No:** US020 **Tasks:** Support multi-language interface. **Priority:** Low **Value Statement:** As a user, I want the application to support multiple languages, so that it is accessible to a diverse audience.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Supported languages must include English, Hindi, and regional options.
2. Language preferences must be savable per user.
3. All interface elements must translate accurately.

#### User Story 21

**User Story No:** US021 **Tasks:** Enable customer tagging. **Priority:** Medium **Value Statement:** As a user, I want to assign tags to customers, so that I can categorize them for targeted services.

**BV:** 500 **CP:** 5

**Acceptance Criteria:**

1. Tags must be customizable and reusable.
2. Tags should appear in the customer profile.
3. Filters for tags must be supported in searches.

#### User Story 22

**User Story No:** US022 **Tasks:** View outstanding dues for customers. **Priority:** High **Value Statement:** As a user, I want to see any outstanding dues for a customer, so that I can prompt timely payments.

**BV:** 500 **CP:** 5

**Acceptance Criteria:**

1. Dues should display amounts, due dates, and associated accounts.
2. Overdue amounts must be highlighted.
3. Payment reminders should be configurable.

#### User Story 23

**User Story No:** US023 **Tasks:** Implement data archival. **Priority:** Medium **Value Statement:** As a user, I want to archive old customer data, so that I can maintain system performance.

**BV:** 200 **CP:** 5

**Acceptance Criteria:**

1. Archived data should remain accessible on-demand.
2. A retention policy must guide data archival.
3. Notifications should prompt users before archival.

#### User Story 24

**User Story No:** US024 **Tasks:** Enable custom dashboards. **Priority:** Medium **Value Statement:** As a user, I want to customize my dashboard, so that I can focus on relevant metrics.

**BV:** 200 **CP:** 8

**Acceptance Criteria:**

1. Widgets must be drag gable and resizable.
2. Preferences must save automatically.
3. Users should have the option to reset to default.

#### User Story 25

**User Story No:** US025 **Tasks:** Implement customer segmentation. **Priority:** High **Value Statement:** As a user, I want to segment customers based on various criteria, so that I can tailor services effectively.

**BV:** 200 **CP:** 3

**Acceptance Criteria:**

1. Criteria must include age, income, and portfolio size.
2. Segments should update dynamically.
3. Reports must be available for each segment.

#### User Story 26

**User Story No:** US026 **Tasks:** Create reminders for follow-ups. **Priority:** Medium **Value Statement:** As a user, I want to set reminders for follow-ups with customers, so that I do not miss important interactions.

**BV:** 500 **CP:** 6

**Acceptance Criteria:**

1. Reminders must support dates, times, and notes.
2. Notifications should appear at scheduled times.
3. Missed follow-ups must be logged.

User Story No: 27  
Tasks: Design login and registration pages, implement authentication, validate input fields.  
Priority: High  
Value Statement:  
As a new user,  
I want to register using my email and phone number,  
so that I can access the application's features securely.

BV: 500 CP: 5  
Acceptance Criteria:

1. Users can register with their email and phone number.
2. Users receive a confirmation email/SMS upon successful registration.
3. Errors for invalid or incomplete inputs are displayed clearly.

User Story No: 28  
Tasks: Build a dashboard interface; connect with the database to display key metrics.  
Priority: High  
Value Statement:  
As a logged-in user,  
I want to view a personalized dashboard,  
so that I can track my progress and see relevant updates.

BV: 500 CP: 8  
Acceptance Criteria:

1. The dashboard displays personalized information based on user activity.
2. Users can navigate to other sections of the app from the dashboard.
3. The design is responsive and user-friendly.

User Story No: 29  
Tasks: Implement search functionality, add filters, optimize results.  
Priority: Medium  
Value Statement:  
As a user,  
I want to search for specific content or features,  
so that I can find what I need quickly and easily.  
BV: 2000   
CP: 13  
Acceptance Criteria:

1. Search results are displayed within 2 seconds.
2. Filters (e.g., by category, date, relevance) can be applied to refine results.
3. Common misspellings in search queries are corrected automatically.

User Story No: 30  
Tasks: Integrate payment gateway, ensure secure transactions, and add invoice generation.  
Priority: High  
Value Statement:  
As a customer,  
I want to make secure payments,  
so that I can complete transactions without worrying about security issues.  
BV: 500  
CP: 5  
Acceptance Criteria:

1. Payments can be made using credit/debit cards, UPI, and wallets.
2. Transactions are encrypted to ensure security.
3. Users receive a digital invoice immediately after payment.

User Story No: 31  
Tasks: Enable notifications, set user preferences, test notifications across devices.  
Priority: Medium  
Value Statement:  
As a user,  
I want to receive notifications about updates,  
so that I can stay informed about relevant activities and alerts.  
BV: 200  
CP:5  
Acceptance Criteria:

1. Notifications can be enabled or disabled by the user.
2. Users receive alerts for critical updates (e.g., upcoming deadlines).
3. Notifications are consistent across devices.

User Story No: 32  
Tasks: Design a user profile page, implement edit functionality, link profile data to backend.  
Priority: High  
Value Statement:  
As a registered user,  
I want to view and edit my profile details,  
so that I can keep my information accurate and up to date.  
BV: 200   
CP: 5  
Acceptance Criteria:

1. Users can view their profile details, including name, email, and preferences.
2. Users can edit fields like contact number, profile picture, and preferences.
3. Changes are saved and reflected immediately.

User Story No: 33  
Tasks: Create a feedback form, connect it to the database, and add email notifications for the admin.  
Priority: Medium  
Value Statement:  
As a user,  
I want to submit feedback about the application,  
so that I can share my experience and suggest improvements.  
BV: 200  
CP: 3  
Acceptance Criteria:

1. A feedback form is accessible from the dashboard or menu.
2. Users receive a confirmation after submitting feedback.
3. Feedback is logged in the database and notified to the admin team.

User Story No: 34  
Tasks: Implement user on boarding, create instructional modals or videos.  
Priority: Medium  
Value Statement:  
As a new user,  
I want to get a guided walkthrough of the application's features,  
so that I can understand how to use it effectively.  
BV: 500  
CP: 5  
Acceptance Criteria:

1. On boarding screens or tooltips guide users through the core features.
2. Users can skip on boarding if they wish.
3. Tutorials are interactive and easy to follow.

User Story No: 35  
Tasks: Develop an activity log, track key user actions, and display a history page.  
Priority: Medium  
Value Statement:  
As a user,  
I want to view my activity history,  
so that I can track my progress and revisit previous actions.  
BV: 200  
CP: 5  
Acceptance Criteria:

1. Activity history includes key actions (e.g., logins, purchases, updates).
2. Users can filter activity by date or category.
3. Logs are securely stored and accessible only to the user.

User Story No: 36  
Tasks: Integrate multilingual support, create language switcher in settings.  
Priority: Low  
Value Statement:  
As a non-English speaking user,  
I want to use the application in my preferred language,  
so that I can interact with it more comfortably.  
BV: 500  
CP: 5  
Acceptance Criteria:

1. The application supports at least 3 additional languages.
2. Users can switch languages seamlessly from settings.
3. Translations are contextually accurate and user-friendly.

User Story No: 37  
Tasks: Add a reward system, track user milestones, and display progress visually.  
Priority: Medium  
Value Statement:  
As a regular user,  
I want to earn rewards for completing specific tasks,  
so that I can stay motivated and engaged with the application.  
BV: 100  
CP: 2  
Acceptance Criteria:

1. Rewards are displayed visually on the user dashboard.
2. Users receive notifications when milestones are achieved.
3. Rewards can be redeemed for discounts, features, or badges.

User Story No: 38  
Tasks: Implement file uploads, validate file types, and integrate cloud storage.  
Priority: High  
Value Statement:  
As a user,  
I want to upload documents or images,  
so that I can share relevant information or keep records.  
BV: 2000  
CP: 13  
Acceptance Criteria:

1. Users can upload files (PDF, JPEG, PNG, DOCX) up to a defined size limit.
2. Files are stored securely and accessible only to authorized users.
3. Errors for unsupported file types are displayed clearly.

User Story No: 39  
Tasks: Add push notifications for reminders, critical updates, and promotions.  
Priority: High  
Value Statement:  
As a user,  
I want to receive timely notifications,  
so that I can stay updated about important events and actions.  
BV: 2000  
CP: 13  
Acceptance Criteria:

1. Notifications are customizable by type (e.g., reminders, updates).
2. Notifications are displayed on mobile and web platforms.
3. Users can opt-out of non-essential notifications.

User Story No: 40  
Tasks: Implement a community forum, enable thread creation, and integrate moderation tools.  
Priority: Medium  
Value Statement:  
As a user,  
I want to participate in a community forum,  
so that I can share knowledge, ask questions, and collaborate with others.  
BV: 200  
CP: 7  
Acceptance Criteria:

1. Users can create new threads, reply to existing threads, and upvote helpful posts.
2. Moderators can review and remove inappropriate content.
3. Forums are categorized by topics for easier navigation.

Document 4: Agile PO Experience

Who is a Product Owner?

A Product Owner (PO) is a key role in Agile and Scrum teams, responsible for defining and prioritizing the product’s features to maximize its value. They act as a bridge between the business, customers, and the development team to ensure the right product is built.

Key Responsibilities of a Product Owner:

1. Understanding Customer Needs – Gathers requirements from stakeholders, customers, and market research to understand what features or improvements are needed.
2. Managing the Product Backlog – Creates, prioritizes, and refines the list of tasks (Product Backlog) that the development team will work on.
3. Defining Clear Requirements – Writes user stories and acceptance criteria to ensure the team understands what needs to be built.
4. Prioritizing Features – Decides which features are most important and should be worked on first, ensuring maximum business value.
5. Working with the Scrum Team – Collaborates with developers, testers, and designers to ensure the product vision is executed correctly.
6. Making Quick Decisions – Provides clarifications, feedback, and approvals during development to avoid delays.
7. Ensuring Business Goals Are Met – Aligns the product development with the company’s strategic goals and customer expectations.

Market Analysis:

To ensure that the **Saksham Application** meets the needs of Axis Bank employees and stands out in the competitive landscape, a thorough **market analysis** is essential. The analysis will cover both the **market need/demand** and the **availability of similar products**.

Enterprise Analysis:

The Enterprise Analysis and Due Diligence on the Market Opportunity for the Saksham Application involves evaluating its feasibility, competitive landscape, and potential for scalability. The application addresses a specific market need, such as healthcare support, financial inclusion, or social welfare. Market research indicates a growing demand for digital solutions in these domains, driven by increasing smartphone penetration, government initiatives, and user awareness. A competitive assessment highlights key players offering similar services, necessitating unique value propositions for differentiation. The scalability of the application depends on factors such as user adoption, regulatory support, technological advancements, and integration with existing systems. Proper due diligence ensures that market risks are assessed, and strategic opportunities are leveraged for sustainable growth.

Product vision and Road Map:

The **Saksham Application** is designed to streamline and optimize the way Axis Bank employees access and manage customer portfolios. By focusing on enhancing productivity, improving customer service, and ensuring real-time access to customer data, the application will serve as a critical tool for internal use in the bank. Here is the vision and roadmap, aligned with the market need analysis

Managing product features:

Managing product features involves effectively **handling stakeholder expectations**, **prioritizing needs**, and ensuring the **alignment of features** with the business goals of Axis Bank. Prioritizing features based on criticality and **Return on Investment (ROI)** ensures that the development team focuses on the most impactful functionalities first, creating value for both the bank and its employees.

Managing product Backlogs:

Managing the **product backlog** involves prioritizing user stories, ensuring that features align with the product vision, and addressing the needs of stakeholders. Reprioritization is an ongoing process that occurs as new feedback and business requirements are received. Here's how to effectively manage the **product backlog** for the **Saksham Application**:

### **Managing Overall Iteration Progress in Agile**

Managing the overall **iteration progress** involves overseeing the progress of each sprint, reviewing the completion of tasks, reprioritizing sprints and epics if needed, and conducting retrospectives to continuously improve the development process. As a **Business Analyst** (BA), I will play a key role in ensuring that the project stays aligned with business goals, stakeholders' expectations, and user requirements.

❖ From this project I have learned how to handle sprint meetings such as

➢ Sprint planning meeting

A Sprint Planning Meeting is held at the start of each sprint to define the sprint goal and determine which items from the product backlog will be worked on. The team collaborates with the Product Owner to break down user stories into tasks, estimate effort, and commit to deliverables, ensuring alignment with business objectives.

➢ Daily scrum meeting

The **Daily Scrum Meeting** (also known as the **Daily Stand up**) is a short, daily meeting where the **development team** discusses their progress. Each member answers three questions: What did I do yesterday? What will I do today? Are there any blockers? It helps ensure the team stays aligned and identifies potential issues early.

➢ Sprint review meeting

A sprint review meeting is held at the end of each sprint to demonstrate the completed work to stakeholders. The team showcases the features developed, gathers feedback, and discusses any challenges encountered. This meeting helps assess progress, refine goals, and make adjustments to the product backlog for future sprints.

➢ Sprint retrospective meeting

A sprint retrospective meeting is held after the sprint review and before the next sprint planning. The team reflects on the just-completed sprint, discussing what went well, what could be improved, and any obstacles faced. The goal is to identify actions to enhance team performance, collaboration, and efficiency in future sprints.

➢ Backlog refinement meeting

A backlog refinement meeting, also known as backlog grooming, is a collaborative session where the team reviews and prioritizes the product backlog. The product owner clarifies backlog items, ensuring they are well defined, appropriately sized, and prioritized. The team discusses requirements, estimates effort, and prepares items for upcoming sprints, ensuring alignment with project goals.

### User Story Example:

**Story No**: US001

**Title**: View Customer Portfolio in Saksham Application

**Description**:  
As an Axis Bank employee, I want to view a customer's portfolio in the Saksham application so that I can quickly assess the customer's financial status and make informed decisions.

### Tasks:

1. Design the UI to display customer portfolio details (account balance, investments, and loans).
2. Implement integration with the backend systems to fetch the portfolio data.
3. Develop a search functionality to filter customer portfolios.
4. Test the portfolio view functionality on various devices and browsers.
5. Ensure data security and compliance with privacy regulations.

### Priority:

* **Priority**: High (as it is essential for employees to quickly access and view customer portfolio details to make timely decisions).

### Acceptance Criteria:

1. The application must display a customer's portfolio details (e.g., savings account, loan status, investments) accurately.
2. The employee should be able to search for a customer using their name or account number.
3. Portfolio details should load within 5 seconds.
4. Only authorized users (Axis Bank employees) can access the portfolio view.
5. Data should be updated in real-time, reflecting any recent changes in the customer’s account.

### BV (Business Value):

* **Business Value**  
  This user story holds high business value as it enables bank employees to access critical customer information quickly, improving decision-making and enhancing customer service.

### CP (Complexity Points)

* **Complexity**  
  Moderate complexity due to the need to integrate multiple backend systems, ensure real-time data accuracy, and comply with security regulations.

In Scrum, a product owner serves as the liaison between multiple areas of an organization. This person communicates with business stakeholders and collaborates closely with Scrum teams to keep all areas of the business informed on a project's development

The product owner develops a vision of a product's function and operation, which in turn allows this Scrum team member to define product features and break those features into product backlog items

Document 5: Product and sprint backlog and product and sprint burn down charts Product backlog

Product Backlog:

A Product Backlog is a comprehensive to-do list that contains all the features, changes, and fixes needed for a product. It serves as the master plan that everyone refers to when building or improving a product, with the most important items placed at the top for immediate attention. The Business Analyst maintains this list by gathering requirements from stakeholders, writing them clearly, organizing them by priority, and making sure everyone understands what needs to be built. As new needs arise or priorities change, the Business Analyst updates this list accordingly, making it a living document that evolves with the product's needs.

| **ID** | **User Story** | **Tasks** | **Priority** | **BV (₹)** | **CP (Fibonacci)** | **Sprint** |
| --- | --- | --- | --- | --- | --- | --- |
| US-01 | User Login (Email & Password) | - Create login UI - Implement authentication - Validate user credentials - Error handling | High | ₹500 | 3 | Sprint 1 |
| US-02 | View Product Catalog | - Fetch product list from DB - Display product images & details - Implement pagination | High | ₹500 | 5 | Sprint 1 |
| US-03 | Add Product to Cart | - Create ‘Add to Cart’ button - Store cart data in session - Update cart UI | Medium | ₹500 | 5 | Sprint 2 |
| US-04 | Secure Payment Processing (UPI, Card, COD) | - Integrate payment gateway - Implement transaction security - Generate order receipt | Critical | ₹2000 | 13 | Sprint 3 |
| US-05 | Order Status Notifications (Email/SMS) | - Configure notification system - Send status updates via Email/SMS - Implement retry mechanism | Medium | ₹500 | 5 | Sprint 3 |

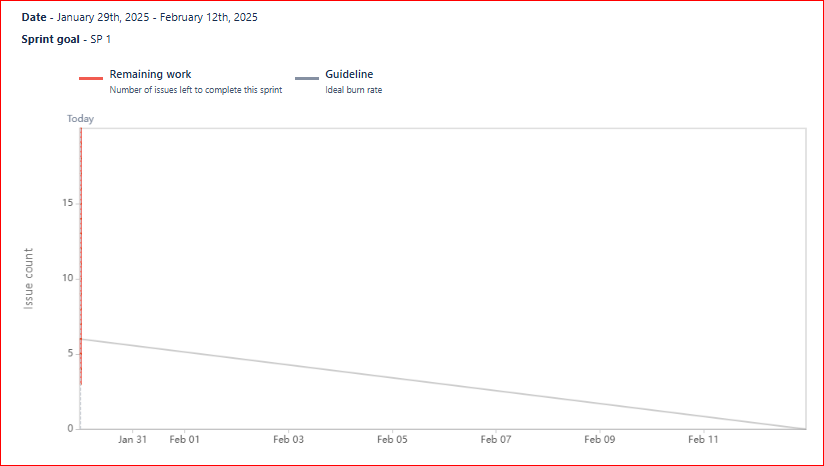
Sprint backlog:

A Sprint Backlog is a smaller, focused list of items selected from the Product Backlog that the team commits to completing during a specific sprint period (usually 2-4 weeks). It represents the detailed plan for delivering the sprint goal and includes the specific user stories, tasks, and technical work needed to create working software. The development team owns this backlog and updates it daily to reflect their progress, with each item broken down into smaller tasks that typically take a day or less to complete. Unlike the Product Backlog, which can change frequently, the Sprint Backlog remains stable during the sprint period, helping the team stay focused and maintain a sustainable development pace while working toward their sprint goal.

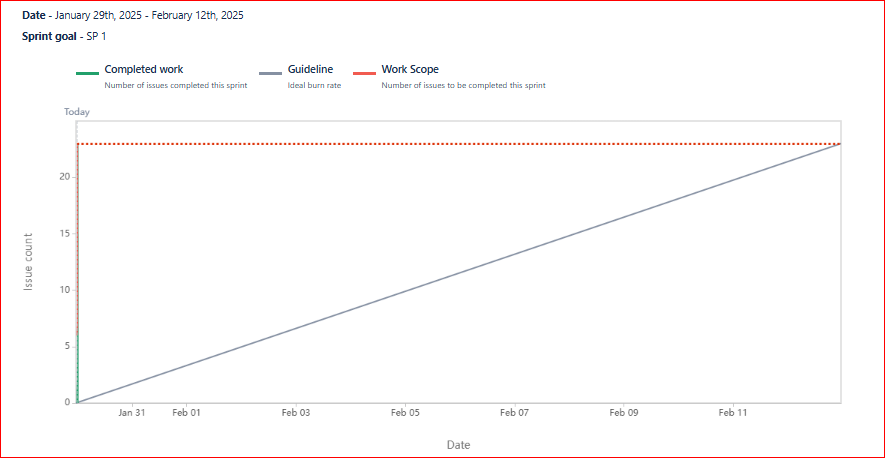
| **ID** | **User Story** | **Tasks** | **Owner** | **Status** | **Estimated Effort (Story Points)** | **Sprint Backlog** |
| --- | --- | --- | --- | --- | --- | --- |
| US-01 | User Login (Email & Password) | - Design login UI  - Implement authentication  - Validate user credentials  - Error handling | Developer A | In Progress | 3 | Sprint 1 |
| US-02 | View Product Catalog | - Fetch product list from DB  - Display product images & details  - Implement pagination | Developer B | To Do | 5 | Sprint 1 |
| US-03 | Add Product to Cart | - Create ‘Add to Cart’ button  - Store cart data in session  - Update cart UI | Developer C | In Progress | 5 | Sprint 2 |
| US-04 | Secure Payment Processing (UPI, Card, COD) | - Integrate payment gateway  - Implement transaction security  - Generate order receipt | Developer D | To Do | 13 | Sprint 3 |
| US-05 | Order Status Notifications (Email/SMS) | - Configure notification system  - Send status updates via Email/SMS  - Implement retry mechanism | Developer E | Done | 5 | Sprint 3 |

Sprint and Product Burn down chart.

Sprint Burn down:



Product Burn down:



Document 6: Sprint meetings Meeting Type 1:

Sprint planning meeting

A Sprint Planning meeting is a collaborative session where the entire Scrum team (Product Owner, Scrum Master, and Development team) comes together at the start of each sprint to decide what work will be delivered in the upcoming sprint and how they will accomplish it. During this meeting, the Product Owner presents the highest priority items from the Product Backlog; the team discusses each item to ensure shared understanding, determines their capacity and velocity, breaks down selected items into specific tasks, and ultimately commits to a realistic Sprint Goal. This meeting typically lasts up to 4 hours for a two-week sprint, where the team decides both WHAT they will build (by selecting Product Backlog items) and HOW they will build it (by creating a detailed Sprint Backlog with specific tasks), ensuring everyone leaves with a clear direction for the sprint ahead.

Date: 17 January 2025  
Time: 11:00 AM IST  
Location: Conference Room 3A, Bengaluru Office

Prepared By: Rahul Mehta (Scrum Master)

Attendees:

1. Saksham Singh (Product Owner)
2. Sai Kiran (Business Analyst)
3. Aman Verma (Developer)
4. Ritu Kapoor (Developer)
5. Karan Desai (Tester)
6. Kavita Nair (UI/UX Designer)

Observers:

1. Ananya Iyer (Quality Assurance Head)
2. Rohan Joshi (Project Manager)

Agenda Topics:

| Topic | Presenter | Time Allotment |
| --- | --- | --- |
| 1. Welcome and Meeting Objectives | Rahul Mehta | 5 minutes |
| 2. Review Sprint Goal and Priorities | Saksham Singh | 10 minutes |
| 3. Discuss User Stories for Saksham Application Enhancements | Priya Sharma | 20 minutes |
| 4. Task Allocation and Dependencies | Aman Verma | 15 minutes |
| 5. Sprint Backlog Finalization | Entire Team | 15 minutes |
| 6. Open Discussion and Q&A | Rahul Mehta | 10 minutes |

Primary Goal:  
Plan the upcoming sprint for the Saksham Application, focusing on enhancing features to allow seamless viewing of the entire customer portfolio.

Other Information

* Resources Needed:
  + Access to Jira for sprint planning and task allocation.
  + Updated user stories from the Product Owner.
  + Customer feedback reports from the previous sprint.
  + A projector for presenting sprint goals and backlog.
* Special Notes:
  + Ensure that the backlog includes any high-priority bug fixes raised in the last retrospective.
  + Align dependencies with external teams (e.g., API integration team).
  + Time box discussions to maintain focus and efficiency.
* Key Observations:
  + Observers are encouraged to provide inputs only during the open discussion segment to keep the planning process streamlined.
  + Document any blockers raised during the meeting for immediate resolution.

Sprint Review Meeting:

### A Sprint Review meeting is a demonstration-focused session held at the end of each sprint where the development team presents the completed work to stakeholders, product owner, and any interested parties. During this informal meeting, the team showcases the working features they've built, collects feedback on the delivered functionality, discusses what went well and what didn't, and gathers insights that will help shape future development priorities. The Product Owner explains which Product Backlog items have been completed and which have not; while stakeholders provide real-time feedback and suggestions, making this meeting, a crucial opportunity for transparency, alignment, and course correction to ensure the product is evolving in the right direction. This session typically lasts for a maximum of 4 hours for a one-month sprint, fostering open dialogue between those who build the product and those who will use it.

**Meeting Type:** Sprint Review

**Date:** 18th January 2025  
**Time:** 3:00 PM IST  
**Location:** Conference Room 3A, Bengaluru Office

**Prepared By:** Rahul Mehta (Scrum Master)

**Attendees:**

1. Saksham Singh (Product Owner)
2. Sai Kiran (Business Analyst)
3. Aman Verma (Developer)
4. Ritu Kapoor (Developer)
5. Karan Desai (Tester)
6. Kavita Nair (UI/UX Designer)
7. Ananya Iyer (Quality Assurance Head)
8. Rohan Joshi (Project Manager)

### **Agenda Topics**

| **Topic** | **Presenter** | **Time Allotment** |
| --- | --- | --- |
| 1. Welcome and Objectives | Rahul Mehta | 5 minutes |
| 2. Sprint Status Update | Priya Sharma | 10 minutes |
| 3. Things to Demo | Aman Verma, Ritu Kapoor | 20 minutes |
| 4. Quick Updates | Team Members | 10 minutes |
| 5. What is Next? | Saksham Singh | 10 minutes |

### **Details**

#### **Sprint Status Update**

* Priya Sharma will provide a summary of completed tasks, any unfinished backlog items, and the overall sprint status.

#### **Things to Demo**

1. **New Features Implemented:**
   * Enhanced customer portfolio view functionality.
   * Improved UI/UX for portfolio navigation.
2. **Bug Fixes:**
   * Resolved login inconsistency issues.
   * Fixed portfolio-loading delays.

#### **Quick Updates**

* Brief inputs from each team member on their key contributions, challenges faced, and learnings from this sprint.

#### **What is next?**

* Saksham Singh will outline priorities for the next sprint, focusing on adding advanced filtering options in the portfolio view and integrating additional customer data points.
* Identify dependencies and blockers for upcoming tasks.

### **Special Notes**

* Ensure that all feedback from the Product Owner is documented for retrospective discussions.
* Observers are welcome to share their insights after the demo section.
* Team members must update Jira with their sprint progress before the meeting.

Sprint Retrospective meeting

A **Sprint Retrospective Meeting** is a review session held at the end of a sprint where the Scrum team reflects on the sprint's progress, identifies successes, discusses challenges, and plans improvements for future sprints. The meeting focuses on what went well, what didn’t go well, and what actions can be taken to enhance efficiency and teamwork. The goal is to foster continuous improvement and optimize future sprints by implementing actionable insights.

### **Sprint Retrospective Meeting Agenda**

**Meeting Type:** Sprint Retrospective

**Date:** 19th January 2025  
**Time:** 4:00 PM IST  
**Location:** Conference Room 3A, Bengaluru Office

**Prepared By:** Rahul Mehta (Scrum Master)

**Attendees:**

1. Saksham Singh (Product Owner)
2. Sai Kiran (Business Analyst)
3. Aman Verma (Developer)
4. Ritu Kapoor (Developer)
5. Karan Desai (Tester)
6. Kavita Nair (UI/UX Designer)
7. Ananya Iyer (Quality Assurance Head)
8. Rohan Joshi (Project Manager)

### **Agenda**

| **Topic** | **Presenter** | **Time Allotment** |
| --- | --- | --- |
| 1. Welcome and Purpose of Meeting | Rahul Mehta | 5 minutes |
| 2. What Went Well | Entire Team | 10 minutes |
| 3. What Didn’t Go Well | Entire Team | 10 minutes |
| 4. Questions and Discussions | Entire Team | 15 minutes |
| 5. Action Items and Next Steps | Rahul Mehta | 10 minutes |

### **Details**

#### **What Went Well**

* Team members will share successes from the sprint, including:
  + Completion of the **enhanced customer portfolio view**.
  + Effective collaboration between developers and the QA team for bug resolution.
  + Timely demo preparation and positive feedback from the Product Owner.

#### **What Didn’t Go Well**

* Discuss challenges faced, such as:
  + Delays in API integration with the customer database.
  + Overlapping tasks causing temporary resource bottlenecks.
  + Communication gaps during sprint planning.

#### **Questions and Discussions**

* Address unresolved questions, including:
  + How can dependencies with external teams be managed more efficiently?
  + What can be improved in terms of sprint planning and backlog prioritization?

### **References for Saksham Application**

* Key focus areas for improvement:
  + Optimizing customer portfolio loading speed further.
  + Enhancing error handling during portfolio updates.
* Review Jira tasks and sprint metrics for deeper insights into team performance.

### **Special Notes**

* Ensure all team feedback is documented for future process improvement.
* Summarize key takeaways and action items at the end of the meeting.

Daily Stand up meeting:

A Daily Stand-up Meeting is a short, time-boxed meeting (typically 15 minutes) where the Scrum team synchronizes their work and discusses progress. Each team member answers three key questions: What was accomplished yesterday? What will be worked on today? Are there any blockers or impediments? The meeting helps maintain transparency, align the team, and identify obstacles early to ensure smooth progress toward sprint goals.

Daily Stand-Up Meeting Agenda

Meeting Type: Daily Stand-Up  
Week: Week 3 (from 15-01-2025 to 21-01-2025)

| Question | Name/Role | Monday (15-01-2025) | Tuesday (16-01-2025) | Wednesday (17-01-2025) | Thursday (18-01-2025) | Friday (19-01-2025) | Saturday (20-01-2025) | Sunday (21-01-2025) |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| What did you do yesterday? | Aman Verma (Developer 1) | Implemented the basic customer portfolio view. | Enhanced API integration for customer data retrieval. | Debugged API integration issues. | Finalized code for the portfolio display feature. | Fixed minor UI bugs in portfolio navigation. | N/A | N/A |
|  | Ritu Kapoor (Developer 2) | Designed the UI mock-ups for the portfolio page. | Developed the portfolio navigation flow. | Worked on responsive design for the portfolio page. | Assisted with API integration testing. | Completed design tweaks based on demo feedback. | N/A | N/A |
|  | Karan Desai (Developer 3) | Verified test cases for API integration. | Conducted regression testing. | Tested portfolio-loading speed and shared feedback. | Validated end-to-end functionality for the demo. | Reviewed bug fixes and updated test cases. | N/A | N/A |
| What will you do today? | Aman Verma (Developer 1) | Enhance error handling for API calls. | Work on data caching for faster loading. | Assist QA with portfolio testing. | Address feedback from the demo session. | Plan next sprint backlog items. | N/A | N/A |
|  | Ritu Kapoor (Developer 2) | Begin coding advanced filtering options. | Test filtering features with sample data. | Fix CSS issues reported by QA. | Review UI alignment post-demo. | Document feature design for the next sprint. | N/A | N/A |
|  | Karan Desai (Developer 3) | Write additional test cases for filtering. | Automate repetitive test cases. | Collaborate with developers on testing bug fixes. | Finalize test results for retrospective. | Update testing metrics for sprint report. | N/A | N/A |
| What (if any) is blocking your progress? | Aman Verma (Developer 1) | Dependency on QA for test results. | Issues with third-party API response delays. | Waiting for feedback on error handling changes. | N/A | N/A | N/A | N/A |
|  | Ritu Kapoor (Developer 2) | N/A | Delay in receiving assets from UI/UX team. | N/A | N/A | N/A | N/A |  |
|  | Karan Desai (Developer 3) | Incomplete test environment setup. | Lack of clarity on edge cases for filtering. | Dependency on Dev team for bug fixes. | N/A | N/A | N/A | N/A |

Special Notes:

1. Developers should ensure their updates are brief and focused (1-2 minutes per person).
2. The Scrum Master for resolution post-meeting should note all blockers raised.
3. Weekend entries are marked N/A unless the team is working overtime or on-call.