



# RFP | Customer Relationship Management (CRM) for Ubank

## Annexure A - Technical Requirements

**U Microfinance Bank Limited (U Bank)**

## Contents

1. Scope of Work.....	2
2. Functional Requirements .....	5
2.1 - Unified Customer Profile and 360-Degree View.....	5
2.2 - Complaint & Case Management.....	6
2.3 - SLA and Escalation Management .....	9
2.4 - Multi-Channel Interaction & Self-Service .....	11
2.5 - User Interface & Usability.....	12
2.6 - Reporting and Dashboards.....	14
2.7 - Administration & Configuration .....	15
2.8 - Reporting & Analytics.....	16
2.9 - AI Features .....	19
3. Non-Functional Requirements.....	25
3.1 - Performance & Scalability .....	25
3.2 - Security.....	25
3.3 - Reliability & Availability .....	26
3.4 - Compatibility.....	27
3.5 - Compliance & Regulatory .....	27
4. Integration Points .....	29
4.1 - IVR/Telephony System Integration .....	29
4.2 - Core Banking System (CBS) Integration .....	29
4.3 - Data Lake / Data Warehouse Integration .....	30
4.4 - Mobile Banking Apps, Public Website and IMB Portal Integration .....	31
4.5 - Email, SMS Gateways & Omni Channel Social Media .....	31
4.6 - External Complaint Portals .....	32
5. Licensing Requirements .....	32
6. General Technology and Information Security Requirements.....	32
7. Service Level Agreement (SLA).....	35
8. Glossary .....	38

## 1. Scope of Work

The successful bidder will be responsible for the end-to-end delivery of their proposed **Customer Relationship Management (CRM)** with integrated **Compliant Management System (CMS)** features. This includes, but is not limited to, the following key areas:

### 1. Through Requirement Gathering and Analysis:

- a. Complete due diligence of incumbent CRM and CMS to ensure that all current processes, functionality (in addition to all the requirements of the RFP) are carried forward and migrated to the new CRM.
- b. Complete due diligence of Ubank Call Center and Customer Resolution Unit (CRU) to capture all the current workflows, Standard Operation Procedure (SOPs) and method of working.
- c. Conduct workshops with Ubank stakeholders and relevant teams for thoroughly gauging the functional and non-functional requirements in accordance to the information provision in the RFP.

### 2. Solution Supply:

- a. Provision of the SaaS based CRM platform **or** on-premises CRM Platform with built-in comprehensive CMS features as per RFP requirements, including all required components such as licenses, software, support services and warranties etc. (except hardware).
- b. **For on-premises solution:**
  - In case an on-premises solution is supplied, complete solution (except hardware) for Primary Site and Disaster Recovery (DR) site must be provided to ensure the mentioned SLA in this document is fully complied with.
  - Bidder must share detailed confirmed specifications for infrastructure requirements (may include, but not limited to: CPU, Memory, Storage, Storage Type, Network, OS, DB etc.) in their proposal for PR and DR sites.
- c. Development of modules for contact/complaint management, inquiry tracking, and lead management, ensuring all current use-cases (logging complaints, queries, service requests and leads) must be covered.
- d. The proposed solution must be a comprehensive, ITIL compliant and ensures the highest standards of data integrity and security.
- e. Solution must offer an intuitive UI/UX design for web (and responsive mobile) interfaces, including role-based dashboards and navigation/workflows.

- f. Implementation of sentiment analysis, predictive routing, voice transcription, dynamic SLA management, and smart notifications as described in requirements.
- g. Solution must provide agents and relationship managers with **a comprehensive 360 Degree single view** of each customer's profile, accounts, past interactions, and feedback history, facilitating informed service and cross-selling opportunities.

### 3. **Deployment & Configuration:**

- a. Full deployment and configuration of the CRM suite, having 360 Degree Customer view with full-fledged CMS features to align with Ubank's call center's operational requirements as per required specifications.
- b. Configuring the CRM with the necessary business rules for ticket routing, approvals, escalations, and notifications, aligned with U Bank's SOPs and enhanced with AI where applicable.
- c. Incorporation of all required security controls, user access roles, audit logging, and compliance features (data protection, reporting needs) as non-functional requirements specify.
- d. Ticketing solution with one-click ticket creation with auto-population of customer context (i.e., from various integrated sources such as (but not limited to) Core Banking.

### 4. **Data Migration:** Migration of existing customer interaction records (complaints, leads, etc.) from the current/incumbent CMS into the new CRM to ensure continuity and historical reference.

### 5. **Integration Services (such as but not limited to):**

- a. Seamless integration with Ubank's existing Active Directory (AD) and LDAP for user authentication and synchronization.
- b. Building and testing integrations with the IVR/call center telephony system, core banking system (Temenos Transact), branchless banking core (SAP), data lake/warehouse solutions, email/SMS gateways, mobile banking app/portal(s), and any other relevant channels for data exchange.
- c. Must integrate with existing channels (such as the current Internet Mobile Banking website, Ubank website forms, Ubank and Upaisa mobile banking apps etc.) to capture interactions.
- d. Integrations with social media platforms/channels (such as, but not limited to Twitter, LinkedIn, Facebook, Instagram etc.) to capture customer interactions and complaints.

- e. Integrations with ITSM and other relevant third-party systems as identified during the project lifecycle.
6. **Testing:** Rigorous and comprehensive testing (system, unit, integration, UATs etc.) of the new/proposed system.
7. **Customization & Workflow Design:**
  - a. Comprehensive customization using built-in feature set of the solution and design of workflows for proposed CRM and CMS modules to precisely match Ubank's specific business processes and ITIL best practices.
  - b. Bidder will design user workflow journeys as per Ubank requirements, SOPs and guidelines determined during the discovery phase.
  - c. Customer service interactions (service requests, complaints, inquiries) and leads arising from service contacts.
8. **Training Programs:**
  - a. Development and delivery of comprehensive training programs for training for end-users (call center staff, CRU officers, etc.) and administrators on the new CMS.
  - b. Provision of all necessary training materials.
9. **Documentation:** Provision of complete technical documentation, user manuals, and administrative guides for the implemented CRM solution.
10. **Ongoing and Post-Go Live Support:**
  - a. Provide continuous support and maintenance in case SaaS platform is proposed as per the agreed Service Level Agreement (SLA).
  - b. Provide on-site support in case on-premises solution as per the agreed Service Level Agreement (SLA).
  - c. Provision of warranty & support for the 5 Year time period.
11. **Compliance to regulatory (State Bank of Pakistan) Frameworks:**
  - a. The bidder must ensure that their proposed solution (either on-premises or SaaS) based model must comply with State Bank of Pakistan's (SBP) frameworks for outsourcing.
  - b. In case of SaaS solutions, the proposed solution must be hosted within the geographical boundaries of Islamic Republic of Pakistan. All user data must be retained within Pakistan.
  - c. In case of SaaS solutions, the bidder must comply with SBP's framework on Cloud Outsourcing (can be found at <https://www.sbp.org.pk/bprd/2023/C1-Annix-A.pdf>).

Bidder will need to provide clause-wise compliance and assurance that all conditions of this framework are fully complied with, in consideration of Material Workloads.

## 2. Functional Requirements

### 2.1 - Unified Customer Profile and 360-Degree View

- a) The system should provide a 360-degree Customer View screen that aggregates all relevant customer information in one place. When an agent searches by a customer identifier (CNIC, account number, phone number, etc.), the CRM will display the customer's profile including personal details (name, DOB, etc.), verification fields (e.g. mother's maiden name for security), and a summary of all products and services the customer has with U Bank. This includes:
  - i. Accounts in Branch Banking (traditional accounts), Branchless Banking (mobile wallet/microfinance accounts), and Islamic Banking, with key details like account type, branch, account status, and balances.
  - ii. Card information (Debit/Credit/ATM cards) and their status (active, blocked).
  - iii. Recent transaction history for each account (e.g., last 5 transactions or a selectable date range) to quickly see customer's financial activity.
- b) **Recent interaction history:** a list of the last X (e.g., 5 or 10) customer service tickets (complaints/queries) the customer has lodged in the past, along with their status and resolution. The agent should be able to click on a past ticket to see details if needed.
- c) **Customer Verification Workflow:** For incoming calls where the customer is an Existing-to-Bank (ETB) customer, the CMS shall streamline verification. After pulling up the customer's profile via CNIC or phone number, a Verification section will present key security questions (e.g., DOB, Mother's Name, last 4 digits of account) for the agent to confirm. A "Verified" button will indicate successful verification, upon which the agent can proceed to handle transactions or sensitive information. If verification fails or if the customer is New-to-Bank (NTB, not found in system), the CRM will guide the agent to handle the interaction as a general inquiry or prospect (with a simplified "walk-in customer" profile).
- d) **Contact Logging from Profile:** From the CRM's 360-degree view, the agent should be able to directly initiate a new complaint, inquiry, or lead by clicking an action (e.g., "Log Complaint" or "New Inquiry") that carries over the customer's details into the case logging form. This avoids re-entering customer data and ensures the interaction is linked to the customer's profile.
- e) **Edit/Update Customer Info:** If during a call, the customer provides updated contact information (phone, address) or other details, the agent should have the ability (with proper authorization) to submit those updates. The CRM might integrate an API to the core banking



system to push those changes or at least capture them for back-office update if direct core update is not allowed (this could be logged as a “customer data update” request type).

- f) **View Only for Sensitive Data:** Certain sensitive data (like full account numbers or full credit card numbers) should be masked by default in the profile, with a role-based control if full view is necessary. This ensures compliance with data security (only authorized roles can see or copy sensitive data).

## 2.2 - Complaint & Case Management

- a) **Complaint Ticket Logging:** Users shall be able to log a new Complaint or Service Request via a unified form in the CMS. Key fields include:
  - i. Customer identifier (CNIC or Account or Phone which can be linked to retrieve profile as above).
  - ii. Contact details (if NTB or if different contact number, etc.).
- b) **Customer Interaction selection:** The agent must first select a main category of the issue (e.g., ATM Issues, Loan Inquiry, Account Maintenance, etc.), then a sub-category (specific issue) from the predefined Customer Interaction list. The system must enforce a two-tier selection (main type then sub-type) for proper classification.
- c) **Description/Remarks:** A free-text field for the agent to be available for recording the details of the customer’s issue in narrative form.
- d) **Attachment:** Ability to attach any supporting file or image (if, for example, a customer emailed a screenshot or document, the agent can attach it to the ticket).
- e) **Channel:** The channel through which the complaint was received (defaulted based on how the agent is logging it; e.g., if it’s a call center agent, default “Phone”, but allow change to “Email” or others if they are logging an email request).
- f) **Priority and Severity:** Agents can mark certain issues as high priority (or the system can auto-mark based on rules using its Artificial Intelligence capabilities) which will be used for escalation and SLA purposes.
- g) **Due Date/SLA:** Once the Customer Interaction is chosen, the system should auto-calculate the target resolution time (Turnaround Time) based on predefined SLAs for that issue type (e.g., 3 days for a funds transfer issue, 7 days for a card issue, etc., as per SOP). This due date shall be bound to the ticket.
- h) **Mandatory Field Enforcement:** Based on the selected Customer Interaction (issue type), the CMS shall dynamically enforce mandatory fields specific to that issue. For example, if “Debit Card not Received” is selected, the system might require entering the card request date or tracking number (if applicable). Certain fields must be filled for certain work codes.

Configuration tables will define which fields are mandatory for each work code, and the system will validate upon submission.

- i) **Unique Ticket ID and Tracking:** Every new case (complaint or inquiry) gets a unique reference number. The format may be alphanumeric (e.g., UB2025-xxxx) for easy communication to customers. This ID is used to track the case through its lifecycle. Currently, our ticket ID format includes the month and year to help identify the timeline of each case.
  - Additionally, the following letters are used to nominate the customer type:
  - C – Branchless Banking Customer
  - F – Franchise/Retailer
  - B – Branch Banking Customer
- j) **Case Status Management:** The system shall manage the status of each case. Standard statuses include Open (newly logged), In Progress/Pending (assigned and being worked on), Resolved (solution provided, pending customer confirmation), and closed. Additionally, statuses like Awaiting Customer Response or Escalated could be used if needed. Agents should be able to update the status as the case progresses, with certain status changes restricted to specific roles, such as the CRU team, Call Center Outbound team, and ADC Operations may have the ticket closing rights.
- k) **Assignment & Routing:** Upon creation, a complaint ticket should be automatically routed to the appropriate queue or user group based on the Customer Interaction and predefined responsibility matrix. For example, a “Service Charges Issue” might route to the Branch Banking team’s queue, while a “Funds Transfer issue” routes to the Fraud department’s queue. The system will use the Customer Interaction mapping to assign the ticket to a department (and may use its Predictive Routing AI features to suggest the best individual agent or team). Users can also manually reassign tickets if needed (with appropriate permissions), for example, a supervisor reassigning from one agent to another. Assignment notifications should be sent to the assignee.
- l) **Ticket Processing & Updates:** Assigned teams (back-office departments like Branch Ops, Card Operations, etc., or specialized CRU officers) will process the case. They will update the ticket with their investigation notes and resolution details in the CMS. The CMS must allow multi-user collaboration on a ticket, capturing each update as a timestamped entry (with the user’s ID) in a ticket log/journal visible on the ticket screen.
- m) **Resolution and Closure:** When the responsible team marks a ticket as resolved, they update the ticket status to “Resolved” or a similar state and include a resolution note. The CMS then moves the ticket to the “Ready to Close” queue, from where the auto dialer initiates an outbound recorded call — for example, press 1 if the issue is resolved or press 2 if not resolved. If the customer presses 1, the complaint should be closed. However, if the



customer presses 2, the complaint must be reassigned to the same team or bucket that initially marked it as resolved.

Additionally, a mandatory “Root Cause” field should be implemented in CMS to ensure that, before marking any complaint as “Claim Resolved,” the precise root cause is identified and recorded against each CMS ticket.

- n) **Multi-Channel Response Logging:** In cases where resolution requires communicating back to the customer via a channel (email response, SMS update), the CRM system should facilitate that. For example, an email reply can be sent from the CRM’s CMS ticket screen, and it will be logged in the ticket history. Similarly, if an SMS or in-app notification is sent to the customer (for simple updates like “Your issue has been resolved”), that should be logged.
- o) **Customer Interaction Tagging for All Interactions:** Even for inquiries or calls that do not result in a formal complaint, the CMS should allow agents to tag the call with a work code. This is a current practice so that every call is categorized (even if just an informational query). The system should have a lighter “Contact Log” function where agents can record the nature of each call (work code, brief note) without creating a full complaint ticket, for tracking purposes. This ensures data is captured for all interactions, feeding into analytics (like identifying top call topics, not just complaints).
- p) **Query/Inquiry Logging:** Not all customer contacts are complaints; some are general queries or requests for information. The CRM must provide a similar form for queries, which may have a slightly different flow (possibly resolved in real-time by providing information). The system should mark these as “Query” type cases, possibly with a simpler lifecycle (open and close on the call itself). However, if a query requires follow-up, it can be escalated to a complaint in the integrated CMS or assigned for research to any specific relevant team. The fields for a query may differ (and be defined by the bank) but should similarly enforce relevant mandatory fields based on the topic.
- q) **Lead Management:** When an interaction yields a potential business opportunity (for example, a customer says they are interested in a loan product, or a non-customer calls to ask about account opening), the agent shall log it as a Lead in the CRM system. The system will provide a dedicated lead capture screen or form with fields like: Customer/prospect name, contact info, product interested in, and any remarks. The agent can assign the lead to the appropriate department or branch (e.g., Sales team or a specific branch manager) directly from the CRM system. The CRM system should provide functionality to track the status of leads (e.g., New, Contacted, In Process, Converted, Closed/Lost) and allow sales staff to update outcomes. This ensures that inquiries of a sales nature are not dropped and are followed through.
- r) **Multi-Channel Intake Integration:** In addition to agent-driven logging, the CRM system and the integrated CMS should be capable of capturing cases, leads and queries directly from

other channels and if a new source channel needs to be added in CMS, it should be made configurable rather than requiring changes at the development level.

- a. **Email:** If a customer sends an email to a designated support address, the CMS should automatically create a ticket with the email contents, categorize it as an email channel contact, and queue it for an agent.
- b. **Public Website:** Complaints or requests submitted via U Bank’s website should feed directly into the CMS with necessary details.
- c. **SMS:** The customers may use SMS short codes for complaint registration; those messages are to be ingested and logged into as tickets with the text content.
- d. **IVR:** The customers may use the IVR self-service to log certain requests (like card blocking). Tickets must be created automatically accordingly.
- e. **Ubank and Upaisa Applications & IMB:** Customers may log complaint through digital channels (e.g., mobile apps, IMB web portal), it must be directly registered in the CMS without manual intervention.
- s) **IVR Call Linking:** Each phone call received at the Ubank call center should be linked to any ticket that results from it. The CMS should store the Call ID or recording ID from the telephony system in the ticket metadata. This way, the voice recording can be accessed later for quality review directly from the CMS ticket.
- t) **Bulk Actions:** For administrators or power users, the CMS could allow certain bulk operations, such as updating the status of multiple tickets (e.g., closing a batch of tickets that were resolved by a general fix) or reassigning all tickets of a certain type to a new team (if a staff member is on leave, etc.). These actions should be permission-controlled to prevent misuse.
- u) **Cancellation/Deletion Controls:** Once a ticket is logged, it cannot be deleted for audit reasons. If a ticket was created in error or a duplicate, users can mark it as “Canceled” or merge it with another ticket, rather than deletion. Only system administrators might soft-delete with proper justification, and even then, an audit record should remain.
- v) **Service Request Fulfillment:** Some customer requests might not be “issues” but standard service fulfillments (e.g., requesting a bank statement, updating address, etc.). The CRM system should handle such requests and the CMS should similarly (as to complaints) log and track for closure. For instance, a request for account statement might be something the agent can fulfill immediately via core banking and mark as closed with outcome “statement sent”.

## 2.3 - SLA and Escalation Management

- a) **Defined Workflows per Category:** The CMS must support the defined processes for each complaint category as per current SOPs. This means when a ticket is assigned to a department, that department’s users have a clear action to take and then forward or update

accordingly. For example, some issues after being addressed by back-office should loop back to call center for closure (as per the ready-to-close concept). The system will facilitate these hand-offs by changing status/ownership automatically once certain actions are taken.

- b) **SLA Tracking:** The system shall track the SLA/TAT for each ticket based on its type and channel. A countdown or due date is visible on the ticket. If a ticket exceeds its SLA without resolution, it should be flagged (highlighted in red or moved to an “Overdue” queue). Managers may view SLA compliance reports (percentage met vs breached). The SLA definitions must be configurable (e.g., 3 working days for funds transfer issues, 7 days for card issues, etc.). The CMS should consider business hours/holidays for SLA calculations.
- c) **Auto-Escalation Rules:** The CMS must support configurable auto-escalation rules based on SLA timelines, customer sentiment, and customer category (e.g., VIP). The system should have the ability to identify VIP customers using predefined criteria such as account type, relationship value, or custom business rules. Once identified, any complaint from a VIP customer must be automatically flagged for immediate attention and escalated to designated management personnel. Escalation actions may include notifying supervisors, elevating ticket priority, or reassigning the case to an escalation team. For example, a rule can be configured to escalate any VIP complaint immediately to the management dashboard, or to escalate tickets that remain unaddressed for 24 hours to the relevant team lead. These rules and VIP criteria should be fully configurable by system administrators.
- d) **Manual Escalation:** Agents and supervisors should have the ability to manually escalate a case to higher authority or to a different team. For example, a supervisor can mark a complaint as Escalated to Tier-2, which might notify the Head of Customer Experience or trigger involvement of a specialist. The system should capture the reason for escalation and the time it was escalated in the ticket log.
- e) **Audit Trail:** Every action in the workflow (creation, assignment, status change, field update, comment added, escalation, closure) must be recorded in an audit trail on the ticket. The audit entry should include timestamp, user, and action details. Audit logs should be read-only.
- f) **Queue Management & Views:** Users will have queue views based on their role:
  - Agents see tickets assigned to them (My Open Tickets) and perhaps unassigned tickets in their team’s queue that they can pick.
  - Department teams may see all the tickets open for their department.
  - Supervisors may see all tickets for their team (or department) plus any escalated ones.
  - A “Pending Customer” queue can list tickets awaiting customer response (so agents can follow up if needed).

- A “Overdue” queue shows breached SLAs.
- g) **Closure Codes and Feedback:** When closing a ticket, the agent should select a resolution code or category (e.g., Resolved Information Provided, Resolved Technical Fix, Resolved Complaint Upheld/Compensated, Withdrawn by Customer, etc.) to categorize outcomes. Also, the system could trigger a customer feedback survey (e.g., send an SMS or email asking for a 1-5 rating or NPS question). If such a survey is conducted, the result should be linked back to the ticket (for reporting on CSAT per ticket).
- h) **Knowledge Base Integration:** The system should integrate a Knowledge Base of help articles or SOP instructions that agents can reference. For example, when working on a ticket, the agent could click “Knowledge” and search for relevant guides (like “How to handle ATM card not working issue”).
- i) **Concurrent Updates Handling:** The system should handle scenarios where multiple users might open the same ticket (e.g., an agent and a supervisor). It should lock records appropriately or merge updates to avoid conflicts (e.g., if one closes while another is editing, there should be a warning or refresh).
- j) **Notifications (Ubank Internal):** Built-in notifications should alert users of workflow events, e.g., when a ticket is assigned to an agent, they get a notification (and/or email). When a ticket is escalated, the target person gets notified. These notifications might appear in the system UI (notification bell icon) and/or via email/SMS depending on user preferences.

## 2.4 - Multi-Channel Interaction & Self-Service

- a) **Telephone/IVR Integration:** When a customer calls the helpline:
  - The IVR should capture basic info (e.g., account or CNIC via keypad or through the customer’s input). When the call is routed to a live agent, the CMS should receive a “screen pop” with the customer’s identifier. The CRM system will then automatically fetch and display the customer’s 360-view to the agent before they answer the call.
  - During the IVR flow, if the customer attempts a self-service (e.g., “Press 1 to block your card”), and the customer ended up needing agent assistance, the system should display the context (e.g., “Card block attempted self-service”) so the agent knows the background without asking the customer to repeat.
  - The CRM should integrate with the telephony system to allow the agent to perform basic call controls from the CRM interface (answer, hang up, hold, transfer), or at least to log the call disposition easily once the call is over.
  - In case of customer requesting a call-backs or scheduling via IVR, such requests should also be fed into the CMS.
- b) **Email Integration:** The CRM system must connect with the customer service email inbox. Emails from customers may trigger the CMS mechanism to create cases automatically. In

such cases, the system should parse the email for customer email address, attempt to match it to an existing customer (if possible via email or info in text) and create a case with the email subject as title and body as description. Attachments in emails should be attached to the case. Agents must be able to reply from within the CRM suite and the reply must get emailed to the customer and appended to the case history.

- c) **Web Portal / Mobile App Integration:** If a customer submits a complaint or query through U Bank’s online banking, mobile app, or public website, that submission should hit an API of the CMS to log the case. Required fields (as per the form) will map to CMS fields. The customer should receive a reference number on submission (which the CMS generates). The CMS will mark the channel as “Web” or “Mobile” accordingly. If the platform allows, customers might check status of their case through the app/portal, which would require the CMS to expose status info via API (an integration point to consider).
- d) **SMS Alerts & Two-Way SMS:** The system must be able to send SMS notifications to customers for certain updates (e.g., “Your complaint [ID] has been resolved. Thank you.”) and also be able to receive SMS for confirmation (for instance, a customer can confirm resolution by replying “Yes”) via integration with SMS gateway.
- e) **Social Media Integrations:** Social media interactions (like queries on Facebook, Twitter, etc.). This means architecturally, the CRM’s APIs should handle inputs from these channels and facilitate case logging within its CMS.
- f) **Privacy and Preferences:** The CMS must respect any customer communication preferences (for example, if a customer opts out of SMS or email notifications, the system should flag that and not send SMS to that number, etc.). This is important as part of compliance and customer experience (no spam or unauthorized contact).
- g) **Internationalization (if needed):** U Bank operates primarily in Pakistan, so the default language is English (with some use of Urdu possibly in correspondence). The CMS UI will be in English; however, the system should allow storing customer names or notes in Unicode (Urdu text) if needed. Using predefined communication/correspondence templates (emails/SMS), multi-language support is required.

## 2.5 - User Interface & Usability

- a) **Modern Web-Based Interface:** The proposed system must have a web-based application feature (accessible via modern browsers) with a clean, intuitive layout adhering to modern UX principles. U Bank’s branding as per guidelines must be implemented for a professional look. Key design aspects:
  - A consistent header or menu with navigation to main sections (Dashboard, Customer Search, New Complaint, Reports, Admin, etc., based on user role).
  - Use of dashboards and summary cards on home screen to give quick insights (for example, an agent’s home screen could show their open tickets count, urgent tickets,

and maybe a sentiment trend; a supervisor's home screen might show team's performance metrics).

- Support for tabbed interface or multi-window: allow users to have multiple tickets open in tabs or search a knowledge article while the ticket stays open, etc., improving multitasking.
  - Provide quick action buttons and keyboard shortcuts for efficiency (e.g., Ctrl+N for new ticket, etc.).
- b) **Responsive and Mobile-Friendly:** The UI should be fully responsive to different screen sizes. On smaller screens (tablet or smartphone), the layout should adjust (stacked elements, hamburger menus) so that supervisors or managers can quickly check dashboards or even agents can log an interaction on a tablet if needed. A dedicated mobile is highly preferred that must be compatible on iOS and Android devices.
- c) **Role-Based Views for CMS:** The content and available actions in the interface will depend on the user's role:
- **Call Center Agent:** Sees the customer search, new case forms, their ticket queue, knowledge base, and basic reports of their performance. They cannot see or change certain admin settings.
  - **Branch Officer:** Similar to agent, but maybe limited to logging new complaints/leads and seeing the status of those lodged by their branch.
  - **CRU/Back-office User:** Sees tickets assigned to their department, has screens to update/resolve tickets, possibly has access to more customer info if needed for investigation.
  - **Supervisor/Team Lead:** Sees all tickets of their team/department, can reassign tickets, view team performance dashboard (like average handling time, number of tickets per agent, CSAT per agent), and handle escalations.
  - **Manager/Administrator:** Has access to all tickets across categories, high-level dashboards (NPS, overall volume, etc.), configuration settings (to update work codes, SLA parameters), and user management (creating new user accounts, roles).
- d) **Accessibility & Input:** The design should ensure readability (clear fonts, adequate contrast) and ease of data entry (dropdowns for codes, date pickers for date fields, etc.). Error messages should be user-friendly, guiding the user to fix missing info or incorrect data.
- e) **Profile and Preferences:** Users (agents) should have personal profiles in the system (with name, role, contact info). They should be able to set some preferences, like notification preferences (email vs SMS alerts for new tickets), or maybe theme (light/dark mode if possible).



- f) **Help and Tooltips:** The UI should include tooltips or help icons explaining fields and features, especially for new AI-driven features the staff might not be familiar with (for example, a tooltip on a sentiment score indicator explaining “This is the AI-determined sentiment of the customer’s last interaction”). A user guide or help section should be accessible from the interface.
- g) **Localization of Date/Time:** Display dates and times in local format (PKT time zone by default) and use 24-hour time or AM/PM consistently as per U Bank standard. The system should be able to account for different time zones or daylight changes.
- h) **Printing & Exporting:** The system must be able to portray a printable view of a ticket and the ability to export lists (like an agent might export their open tickets list to Excel if needed). Also, any reports should be exportable to common formats (Excel/PDF).
- i) **Session Management:** The UI should have a timeout for inactive sessions (for security). Also, allow users to securely re-login without losing context if possible (or autosave work on a ticket draft if session times out).
- j) **Error Handling & Feedback:** If an action is successful or fails, the system should clearly notify the user (e.g., “Ticket saved successfully [ID#]” or “Error: unable to connect to server”).

## 2.6 - Reporting and Dashboards

- a) **Real-Time Dashboard:** The system should provide a dashboard interface that updates in real-time for supervisors and managers. This could show stats like number of calls today, number of new complaints logged, complaints resolved today, average resolution time, etc. Graphical elements (charts, graphs) should be used for quick understanding of trends (e.g., a line graph of CSAT scores over the past months, bar chart of complaints by category).
- b) **Standard Reports:** The system should offer a suite of standard reports, accessible via the UI, such as:
  - Open Tickets by Department, with aging.
  - SLA Compliance Report (percentage of tickets closed within SLA per month, broken by category).
  - Top 10 Complaint Categories (which issues are most frequent).
  - Agent Performance Report (tickets handled per agent, average handling time, customer feedback).
  - NPS/CSAT Trend (if surveys are done via the system).
  - Lead conversion report (how many leads were generated and converted, etc.).
  - Channel wise volume report (number of interactions by channel phone, email, branch, chatbot, etc.).

- c) **Ad-hoc Query:** Power users or analysts should be able to extract data via query or filters. The system should provide a query builder or allow filtering on the ticket list (for example, filter tickets by date range, work code, branch, etc., and then export that data).
- d) **Data Lake Feed:** All transactional data (tickets, updates, closures, survey responses, etc.) should be logged to the enterprise data lake in a structured format (possibly near real-time or batch) to allow U Bank's analytics team to run advanced analysis or join with other data (e.g., core banking data) for deeper insights.
- k) **Audit Reports:** There should be reports or extracts to show audit trail, for compliance. For instance, "All actions on Complaint ID X" or "All tickets created/closed by User Y in period Z". A systematic report could be generated for each complaint type, providing clear visibility of the tickets remaining in each bucket.

For example: A ticket with a 7-day TAT remained in the CRU bucket for 1 day, Operations for 3 days, IT for 1 day, and Call Center for 2 days. The report should therefore reflect the actual time-based weightage for each bucket.

i.e., CRU: 15%, Operations: 45%, IT: 15%, Call Center: 25%, accordingly.

- e) **Regulatory Reporting:** If the State Bank or other regulator requires regular reports of certain metrics (like number of complaints received, resolved, pending beyond TAT), the CMS should be able to produce these easily from the data.
- f) **Printable Reports and Charts:** Users should be able to print or PDF the reports and dashboards for meetings or records. Charts should render clearly in printouts as well.
- g) **Drill-down Capability:** On dashboards, allow managers to drill down into details. E.g., clicking on the "Open complaints: 50" might list those 50 below or take the user to the detailed view of those cases.
- h) **Scheduling Reports:** The CRM and CMS should allow scheduling of certain reports to be emailed to specific stakeholders periodically (e.g., a weekly complaint summary to the Chief Customer Experience Officer).

## 2.7 - Administration & Configuration

- a) **Customer Interaction Management:** The system should provide an interface for authorized admins to manage the list of Work Codes (complaint categories and subcategories). They should be able to add new codes, retire old ones, or modify descriptions as the business evolves. This also includes defining which department or user group each code is assigned to, and what the default SLA is for that code.
- b) **User and Role Management:** The system will integrate with the corporate Active Directory/SSO for user authentication and must maintain its own authorization roles. Admins should be able to create new user accounts (or assign roles to imported AD users), assign them to one or multiple roles/teams (e.g., an agent in Call Center Level 1, or a manager in

Complaint Resolution Unit), and deactivate users who leave. Password policies (for local accounts) and multi-factor authentication should be enforced in line with security requirements.

- c) **Templates and Canned Responses:** Admins must be able to manage templates for email/SMS responses and chatbot responses. For example, the content of the SMS sent on resolution, or email templates for certain frequent queries, can be created and updated.
- d) **Master Data Configuration:** Other lookup values like branches list, products list, reasons for closure, escalation matrix contacts, etc., should be configurable in the system settings.
- e) **AI Features Control:** For the AI features provided by the system, the admin users must be able to controls their thresholds (e.g., sentiment score threshold that triggers an alert), ability to retrain or update the predictive models and toggling features on/off if needed during rollout.
- f) **Integration Settings:** Ability to configure API endpoints or credentials for the integrations (IVR, core banking, SMS gateway, etc.) in a secure manner.
- g) **Audit of Config Changes:** Just like transaction audit, any change in configuration (like SLA values, work codes, user roles) should be logged with who did it and when.
- h) **System Maintenance Features:** Admins should have access to a dashboard for system health e.g., last data lake sync time, IVR link status, etc. If an integration fails (like email fetch fails), an alert should be shown to system admins.

## 2.8 - Reporting & Analytics

### 8.A - Operational Reports

- a) **Open Tickets Report:** A report listing all open (unresolved) tickets, with key details (ticket ID, customer name, issue category, date opened, SLA due date, current assignee, status). This report can be filtered by department, priority, age, etc. to help team leads manage workloads.
- b) **Aging Report:** Shows tickets bucketed by age (e.g., 0-2 days old, 3-5 days, >5 days, >SLA, etc.), often separated by category or department to highlight backlogs and SLA breaches.
- c) **SLA Compliance Report:** For a given period (weekly, monthly), the number and percentage of tickets resolved within SLA vs. those that breached SLA, broken down by department and issue type. E.g., “Card Ops: 95% within SLA, 5% breached (details...)”, to identify where processes might be lagging.
- d) **Volume of Interactions Report:** Number of interactions (tickets) logged per day, week, or month. It should be able to break down by channel (how many via call vs email vs branch), by Customer Interaction category, and by branch/region if applicable. A trend line month-over-month to see peaks (e.g., spike during a product launch).

- e) **Top N Issue Categories:** Identify the most frequent complaint types for a given period (top 10 complaints causes, top 5 inquiry topics). The system should be able to rank these. For example, a pie chart of complaint categories by proportion. This could feed into product improvement priorities.
- f) **Agent Workload & Performance:** For each agent (or each team), how many tickets did they handle in the period, average resolution time, reopen rate (how many of their closed tickets were reopened by customers, indicating perhaps incomplete resolution), and customer feedback score.
- g) **First Contact Resolution (FCR):** Report showing the percentage of cases resolved on the first contact (no follow-up needed). This might require defining FCR (e.g., tickets closed within one day and not reopened).
- h) **Call Summary Reports:** Report on number of calls taken, average call duration, abandon rates, etc., possibly more from call center ACD side but CMS could store call counts per agent.
- i) **Lead Funnel Report:** For leads captured in CRM, how many are in each stage (new, assigned, contacted, converted, lost). Also conversion rate: out of leads generated via service interactions, what % became customers or sales.

## 8.B - Analytical & Strategic Reports

- a. **Customer Interaction Management:** The CMS should provide an interface for authorized admins to manage the list of Work Codes (complaint categories and subcategories). They should be able to add new codes, retire old ones, or modify descriptions as the business evolves. This also includes defining which department or user group each code is assigned to, and what the default SLA is for that code.
- b. **Customer Satisfaction (CSAT) Report:** The CRM must perform CSAT surveys post-resolution and aggregate the results in a presentable form and reports. E.g., average CSAT score by month, and distribution. Possibly break down CSAT by issue category or by agent to spot patterns (maybe certain issues always yield lower satisfaction and needs a process fix).
- c. **Net Promoter Score (NPS) Trend:** CRM must conduct NPS surveys periodically, include a chart of NPS over time (and aggregate the results in a presentable form). Also, correlate NPS dips with service metrics (like did NPS drop in a quarter where complaint volumes were high or resolution times spiked?).
- d. **Sentiment Analysis Reports:** Summarize the sentiment outcomes collected by the proposed system's AI:
  - a. Overall sentiment distribution of all calls/chats (e.g., 70% positive, 20% neutral, 10% negative).
  - b. Trend of customer sentiment over time (is the percentage of negative interactions decreasing as we improve service?).

- c. Sentiment by channel (maybe calls are more negative than chats, indicating maybe wait times on calls cause frustration).
- e. **Sentiment by agent:** To identify if some agents consistently have customers leaving happier (could learn from them) or vice versa (target for training).
- f. **Root Cause / Theme Analysis:** Using text analytics on descriptions and transcripts, the system might offer insights on root causes of complaints. For example, cluster analysis might show that many negative interactions mention “mobile app crash”. This is beyond basic reporting, veering into AI insights, but even a simpler tag cloud of frequent words in complaints can be useful. The CMS could support exporting data to such tools.
- g. **Resolution Time Analysis:** A report that shows average resolution time by issue type. E.g., “Account opening issues: avg 2 days, Card issues: avg 5 days, etc.” with min/max.
- h. **Escalation Report:** How many tickets were escalated (and to what level) in a period. If many escalations in a certain category or by a certain agent.
- i. **Complaints per Customer Segment:** As per customer profile segments (individual, enterprise, rural, urban, etc.), the system must report if any segment is having more issues. E.g., “Rural customers lodge complaints 30% more about network connectivity issues” valuable for targeted improvements.
- j. **Regulatory Compliance Report:** A special report form that shows, for all complaints in period, details needed for regulator (like ID, date opened, date closed, days taken, nature of complaint, outcome). This must be exportable in excel and PDF formats.

### 8.C - Dashboard Widgets

For example, a manager’s dashboard might show:

- CRM system info for customers (in line with defined agent /manager view & access privileges)
- Today’s new tickets count vs yesterday.
- Current open tickets (and how many high priority).
- A chart of top categories this week.
- Agent leaderboard by tickets closed.
- Current overall sentiment gauge (like an aggregate of today’s calls).
- SLA compliance rate this month.
- A map or branch list if region matters (like which branch generating most complaints).
- An agent’s personal dashboard might show:
  - Their average CSAT from surveys.
  - Their open tickets and average age.
  - Comparison with team avg resolution time, perhaps to motivate improvement.

### 8.D - Ad-hoc Query and Data Access

For custom analysis, the system must provide functionality to users to perform SQL queries and advanced search where a user can combine filters (e.g., find all “fraud” cases in “Karachi region” between June and July that took > 10 days to close). The results can then be exported.

## 8.E - Visualization and Delivery

- All reports should be viewable online and exportable with charts where appropriate. For tabular data, allow sorting, filtering.
- Reports and dashboards should allow export to common formats (PDF, Excel, CSV etc.).
- The system should allow scheduling of report emails.
- Where applicable, drill-down capability in reports (click on a number to see the list of items comprising it).
- Integration or linking to Ubank's BI Platform from CRM might be considered for advanced analytics. However, the CRM and integrated CMS itself should cover operational needs out-of-the-box.

## 8.F - Data Accuracy and Consistency

- a) Historical reporting: The system should retain data to allow year-over-year comparisons, etc. If archiving older than X years, note that long-term trends might need the archived data loaded to data lake for analysis.
- b) The CRM must enforce data quality (e.g., no missing work code, etc.).
- c) Data should also refresh data frequently if cached.
- d) For heavy calculations (like sentiment trends) use a pre-aggregated table updated nightly.
- e) Ensure time zone consistency in reporting (all in Pakistan local time unless specified).

## 2.9 - AI Features

The CRM and integrated CMS must have inbuilt Artificial Intelligence (AI) features to augment the system's capabilities, providing smarter automation and deeper insights. Below are the requirements for each AI feature:

### a. Real-Time Sentiment Analysis

**Scope:** The CMS shall analyze customer communications (both voice calls and text-based interactions) to determine the customer's sentiment (positive, neutral, negative) in real-time and after interaction completion.

- i. **Voice Call Sentiment:** Utilize speech-to-text and tone analysis on live calls. As the customer and agent speak, CMS's AI engine should evaluate vocal cues (such as tone, pitch, volume) and the transcribed words to gauge the customer's emotional state. The system should update a visible Sentiment Meter on the agent's screen (and supervisor dashboard) during the call. For example, a bar or icon that shifts from green (positive) to red (negative) if the customer becomes upset. This gives the agent



immediate insight and an opportunity to adjust their approach if the sentiment is turning negative.

- ii. **Text Sentiment:** For channels like email or chat, use in-built Natural Language Processing (NLP) to classify sentiment. The AI should consider words used, punctuation, and context. The sentiment result can be a simple label (Positive/Neutral/Negative) or a score (e.g., -1 to +1 or 0-100 scale).
- iii. **Agent Visibility:** The agent handling the interaction can see the sentiment indicator but it should be presented in a helpful manner (e.g., “Customer seems frustrated” or an icon, rather than a confusing number). This can guide the agent to be more empathetic or call a supervisor if needed.
- iv. **Supervisor Alerts:** If sentiment drops below a certain threshold (indicating a customer is very upset or angry), the system should alert a supervisor in real-time.
- v. **Post-Interaction Sentiment Record:** After a call or chat is completed, the final sentiment score and possibly a sentiment timeline should be saved to the ticket.
- vi. **Multiple Participants:** In case of voice interactions, it’s ideal if the sentiment analysis can distinguish between customer sentiment and agent sentiment (to monitor agent tone as well).
- vii. **Accuracy and Tuning:** The sentiment analysis model should be tuned for the context of U Bank (financial services). It should recognize domain-specific cues (e.g., “I’m unable to feed my family because my account is locked” is highly negative, beyond just the words “unable” might imply). False positives/negatives should be minimized. The system might allow feedback, for example, QA can flag if the sentiment was wrongly detected, to improve the model over time.
- viii. **Privacy Consideration:** Customers should be informed (perhaps in terms & conditions) that calls may be monitored and analyzed by AI for quality purposes. This is usually covered under call recording notices.

## b. Predictive Ticket Routing

**Objective:** Automatically route or assign incoming tickets to the most appropriate agent or team using AI predictions, rather than solely static rules. The goal is to reduce resolution time and improve first-contact resolution by matching issues to those best equipped to handle them.

- i. **Inputs for Prediction:** The predictive model should consider various data points:
  - a. **Ticket metadata:** work code/category, priority, channel, keywords in description.
  - b. **Customer profile:** customer’s segment (VIP, new customer, etc.), history of past issues (did it require multiple touchpoints?).
  - c. **Agent skill data:** historical performance of agents on certain issue types (who resolved similar issues fastest or with high CSAT), current load on agents, language skills, etc.
  - d. **Context:** sentiment (if known at creation or from initial customer input) could also influence urgency.

- ii. **Functionality:** When a new complaint is logged, instead of purely going by a predetermined department, the system's AI model will suggest the best route:
  - a. It could suggest a specific agent who has high success with that type of issue.
  - b. Or suggest a specialized queue if it predicts complexity (e.g., issues predicted to be complex go to a Tier-2 team directly).
  - c. The system might also predict if an issue will violate SLA under current conditions and thus mark it as requiring urgent attention or escalation.
  - d. **Automation vs Suggestion:** We can implement either fully automatic routing or a recommendation system. Initially, it could be a recommendation that the supervisor can approve (especially if we want to build trust in the AI). Over time, as the model proves accurate, routing can be auto-applied with override capability.
  - e. **Skill-Based Matching:** Enhance traditional skill-based routing by using AI. For example, if a customer writes in Urdu (detected via NLP), the system can route to an Urdu-speaking agent. If an angry customer is detected, route to an agent with high satisfaction scores.
  
- iii. **Benefits Realization:** The system should aim to achieve:
  - a. **Robotic Process Automation (RPA):** Use RPA bots to handle certain repetitive tasks directly from CRM. For example, if a complaint requires fetching data from an old legacy system that has no API, an RPA bot could automatically do that and update the CMS.
  - b. **Faster Resolutions:** By assigning to the right person first time, avoid reassignments and loopbacks.
  - c. **Balanced Workload:** AI can factor in current agent availability and load, balancing queues better than static round-robin would.
  - d. **Higher First Contact Resolution:** especially relevant if the issue can be resolved on the first call by connecting to the right expert.
  - e. **Learning Loop:** The routing AI should continuously learn from outcomes. For example, if it routed a ticket to Agent A and Agent A resolved it quickly, that reinforces the model. If it routed to Agent B but it got reassigned or took too long, the model adjusts its criteria. Essentially, it uses historical data to improve future decisions.
  - f. **Override and Fallback:** There must be business rule fallbacks. If the AI confidence is low or something unusual about the case, it should fall back to default rule (like route by Customer Interaction to department). Additionally, human supervisors can override routing decisions if needed (e.g., if an agent is unexpectedly out sick, the supervisor can manually redistribute even if AI had assigned some to them).

- g. **Transparency:** It's helpful if the system provides a rationale for its routing (for trust). For instance, show a note: "Auto-assigned to Card Ops Team (confidence 92%) based on issue type and past resolution data."
- h. **No Delay:** The prediction should happen quickly (<1 second ideally) as part of ticket creation. It should not noticeably slow down the logging process.

### c. Voice-to-Text Transcription with Emotion Tagging

- i. **Call Transcription:** All voice calls should be transcribed to text by the system, either in real-time (streaming transcription) or shortly after the call (batch transcription). The requirement is to have a text record of what was said by the customer and agent. This text should be attached to the interaction record.

Real-time transcription can be displayed to agents (especially for complex calls, this might not be necessary to show live but could help hearing-impaired agents or to quickly catch details like numbers if they missed them).

- ii. **Post-call transcription:** Once call ends, within a couple of minutes, the full transcript is available in the ticket.
- iii. **Emotion Tagging:** As part of speech analytics, tag the transcript with emotions or sentiment at different points. For example:
  - a. Mark sentences or segments where the customer sounded angry, sad, or happy.
  - b. The transcript could have annotations like "[Angry tone] I've been waiting for an hour!" or simply metadata like line 10: sentiment -0.75 (negative).
  - c. This helps review calls quickly a supervisor could jump to the sections where emotion was high. Or the system could automatically highlight "emotion spikes."
- iv. **Accuracy:** Use advanced speech recognition to handle various accents and languages that U Bank encounters. The system should accurately capture important details like names, account numbers (which might require a phonetic library or asking agent to confirm numbers), and amounts. Acoustic models might need to be customized for local accent or banking terms.
- v. **Storage and Search:** Transcripts should be stored in a text-searchable way. The CMS could allow keyword search across transcripts. For instance, searching all calls for the phrase "not received card" might identify a trend.
- vi. **Quality Monitoring:** The combination of transcript + emotion tagging can revolutionize QA. QA personnel can read transcripts instead of listening to every call, which is faster. They can also get an "Emotion timeline" of the call to focus on critical parts. For instance, if a call started calm and turned angry in the middle, the QA can see that and investigate why (maybe the agent said something).

- vii. **Training AI & Compliance:** Transcripts can be used to train the sentiment model further, and also they act as a compliance record to prove what was communicated. If a dispute arises (“the agent promised me X”), having the transcript makes it easier to verify.
- viii. **Data Privacy:** Since calls are being converted to text and stored, that data must be protected similarly to any written customer data. Also, if sensitive info is spoken (like passwords or card numbers), the transcription service should ideally mask or not keep those (or the agent should pause recording when asking for PINs).
- ix. **Multilingual Support:** If customers speak in languages other than English (Urdu or mixed), the transcription engine should handle it or at least capture in original language. It might transcribe Urdu spoken in Roman script or possibly integrate with an Urdu speech model. This might be a stretch goal depending on technology availability.
- x. **Integration with Sentiment:** The emotion tagging on voice is essentially an aspect of sentiment analysis but specifically derived from voice signals (tone, pitch). This complements the text sentiment. Even if words are polite, the tone might be angry; the system should catch that. So we should combine both cues for overall sentiment scoring.

#### d. Visual Analytics Dashboards

- i. **Interactive Dashboards:** The system should have graphical, interactive dashboards for various metrics (as described in Reporting). AI can augment these by highlighting trends or anomalies automatically. For example, if complaint volume in a category spikes 300% this week, the dashboard could flag that insight (“anomaly detected: unusual spike in ATM complaints compared to last 4 weeks”).
- ii. **NPS/CSAT Trends:** the CMS should chart NPS over time (monthly, quarterly) and correlate with data (e.g., “NPS dipped in Q2 possibly due to longer call wait times”). Using AI, the system might identify key drivers from text feedback (text analytics on survey comments).
- iii. **Agent Performance:** Use data to potentially rank or rate agent performance with more nuance. Instead of just average handle time, incorporate sentiment outcomes (e.g., Agent A has high resolution rate but often low sentiment calls meaning maybe he resolves but customers leave unhappy; Agent B has slightly longer times but customers are happy). Such insights can be drawn via AI analyzing the intersection of sentiment, resolution, and survey scores per agent.
- iv. **Predictive Analytics:** The dashboards might include some forward-looking statements, like forecasting workloads (“Based on trends, next month’s call volume is expected to increase by 10%”) or highlighting customers at risk (if churn model is in place, show how many high-risk customers called in).

#### e. Enhanced Escalation & Notifications (Smart Notifications)

- i. **Intelligent Escalation Triggers:** Beyond static rules, use AI to determine when to escalate. E.g., if a normally low-priority issue has a very angry customer (sentiment-

driven), escalate it. Or if a VIP customer (determined by profile or ML identifying high lifetime value) calls with any complaint, automatically flag it to head of customer service.

## ii. **Notification Types:**

- a. **SLA Breach Warnings:** Notify responsible agent and their supervisor X hours before a ticket is due if it's not yet resolved. Possibly increase frequency as deadline nears.
  - b. **High Priority Alerts:** Notify relevant managers immediately when a high-priority ticket is logged (like fraud, or CEO/Board member's relative calls, etc.).
  - c. **Queue Alerts:** If backlog in a queue grows beyond threshold, send an alert to reallocate resources.
  - d. **Sentiment Alerts:** Notify on negative sentiment in live calls.
  - e. **Milestone Updates:** For a long-running case, notify the customer relationship manager or branch manager when it's resolved so they can follow up personally if needed (especially for important clients).
- iii. **Channel of Notification:** Internal notifications can appear within CRM and CMS (notification center) and via email.
  - iv. **Consolidation and Smart Filtering:** "Smart" notifications mean avoiding spamming. The CMS should consolidate notifications when possible (e.g., instead of 10 separate emails for 10 tickets breaching at 6pm, one summary email can be sent).
  - v. **Reminders for Pending Customer Input:** If a ticket is waiting on customer (e.g., more documents from them), the system can remind the agent to follow up.
  - vi. **Learning Preferences:** If the system notices a user consistently ignores certain notifications but responds to others, it might suggest adjusting preferences (or automatically adjust frequency).

## f. **AI for Quality and Training**

- i. **Quality Scoring:** Use speech analytics on call recordings to score calls for compliance (e.g., did agent use proper greeting, did agent offer upsell if required, etc., can be detected by keywords).
- ii. **Training Recommendations:** If an agent is struggling in a certain type of query, the system could recommend training modules to them or to their supervisor.
- iii. **Knowledge Base Gaps:** AI analyzing tickets might find frequently asked questions that don't have a good knowledge article (because agents keep writing custom answers). It could flag these so the content team can create new help articles or chatbot answers.

## 3. Non-Functional Requirements

### 3.1 - Performance & Scalability

- a) **Response Time:** The system should be highly responsive. Common operations (opening a customer profile, saving a ticket, loading a dashboard) should be instantaneous under normal and high loads. The UI should be designed to load data asynchronously where possible to avoid locking the interface. The details for response times are elaborated in the SLA section of this document.
- b) **Throughput, Capacity & Concurrent User Licenses:** The system will be used by the U Bank call center (dozens of agents concurrently) and branch staff across the network. It should be able **to scale up to 2-3 times in the future as the bank grows** or if more channels/users are onboard.
- c) **Scalability:** The service architecture should be scalable horizontally to handle increased workload, such as a surge in interactions or addition of new channels. It must be able to handle peak volumes, for example during promotions or service outages when call volume might spike. The solution must be horizontally scalable to incorporate any amount of increase in the concurrent no. of users.
- d) **Batch Processing Windows:** In case of any batch jobs are run (like nightly data sync to the data lake, or model training jobs), they should be designed to complete within non-peak hours and not interfere with daytime operations. Any long-running tasks should be asynchronous and not lock user-facing functions.
- e) **Extensibility for New Features:** The CRM system should be open to adding new modules or features later, such as integration of new communication channels (social media, WhatsApp), or new AI features, without needing a complete redesign. Using standard protocols (REST APIs, etc.) for integration ensures easier addition of future components.

### 3.2 - Security

- a) **Authentication & Authorization:** Only authorized users may be able to access the CRM and its inbuilt features of CMS. Integration with U Bank's Single Sign-On (SSO) or Active Directory for user login is required, enforcing strong authentication with two-factor security. Role based Access Controls (RBAC) must be implemented.
- b) **Data Encryption:** Data in transit and in rest must be encrypted (HTTPS for web access, secure API calls with TLS). For storage data in rest (such as customer sensitive data (PII) in the database), it must be encrypted or hashed as appropriate.
- c) **Data Privacy Compliance:** The system must comply with data protection regulations (e.g., Pakistan's PDPA if applicable or general best practices). Customer data should only be used



for intended purposes. Access to full customer information might be restricted for certain roles. Anonymization or deletion of customer data should be possible if a customer exercises a right to be forgotten (if applicable by policy).

- d) **Audit & Compliance:** The system must maintain a detailed audit log of user activities for at least a period of 5 Years. These logs must be tamper-proof. Administrative actions (like changing an SLA or deleting a ticket) should require higher privileges and must also be logged.
- e) **Session Security:** User sessions should forcefully be timed out after a period of 10 minutes of inactivity (or as per Ubank policy) to prevent unauthorized use. The system should have protections against common web vulnerabilities (SQL injection, XSS, CSRF attacks, etc.). Use of a modern web framework and security testing is expected.
- f) **Access Control by Network:** Ability to restrict access to CRM system via U Bank's internal network only.
- g) **Backup and Recovery:** Regular backups of the CRM system and all CMS data must be taken and maintained in encrypted form by the service provider. The service provider must perform restoration within least recovery time in case of failure as per SLAs defined in this document.
- h) **Failover & Redundancy:** The service provider must ensure that the system should be deployed in a redundant setup with no single point of hardware or software failure. If the primary data center fails, the service provider must use their secondary / DR (Disaster Recovery) site to restore service within the bounds of SLA defined in this document.
- i) **Penetration Testing:** Before go-live, the system must undergo security testing (vulnerability scan and penetration testing) to ensure no critical vulnerabilities exist. Any issues found must be remediated.
- j) **Compliance Standards:** Align with any specific standards U Bank adheres to, such as PCI DSS (if card data is involved), ISO 27001 for ISMS, or SBP's IT Governance guidelines for banks. PCI compliance would be necessary.

### 3.3 - Reliability & Availability

- a) **Availability:** The CMS should be available to users whenever needed. U Bank's call center likely operates extended hours or 24/7. Target SLA components are mentioned in the SLA section of this annexure. Any maintenance windows should be planned in off-peak hours and communicated, and ideally zero-downtime deployment techniques are used for updates.
- b) **Error Tolerance:** The system should handle expected error conditions gracefully. For example, if any integrated system is down, the CRM and CMS should not crash; instead, it might warn the agent "Account details currently unavailable" but still allow the complaint to be logged. Similarly, if one microservice (e.g. sentiment analysis) fails, it should degrade gracefully (log the call without sentiment rather than blocking the entire process).

- c) **Data Integrity:** Ensure transactional integrity in the database. Use proper commit/rollback so that partial updates do not occur (e.g., if a ticket is being saved and the system crashes, it shouldn't create a half-record). Also, consistency between integrated systems (like core banking and CRM/CMS) should be maintained.
- d) **Monitoring & Alerts:** The service provide must facilitate Ubank's IT and Information Security teams for monitoring the CRM system (inclusive of its CMS module). If something goes wrong (like integration failure, unusually slow response, high error rates), automated alerts from service provider must be shared with Ubank teams.
- e) **Supportability:** Provide adequate logging inside the application for debugging issues. This includes logs for API calls, background job processing, etc., which can be turned on for troubleshooting. Also ensure there's a mechanism to capture user feedback or error reports if something goes wrong on the front-end (like a user can report an issue or the system sends an automatic error report).

### 3.4 - Compatibility

- a) **Browser Compatibility:** The web application should support the latest versions of common browsers used in U Bank's environment (such as Chrome, Edge, Safari, Firefox etc.). It should degrade gracefully on older versions or at least clearly notify if a browser is not supported.
- b) **OS & Device Compatibility:** The proposed system must be compatible with Windows PC and Macintosh machines and their latest associated operating systems (Windows 11 and MacOS Tahoe). However, the system must be backward compatible with older variants of these operating systems.
- c) **Screen Resolution:** Design for at least 1366x768 and above (common laptop resolution) while being responsive to higher resolutions (Full HD and 4K monitors used by supervisors) and devices with lower resolution (such as tablet screens).
- d) **Integration Compatibility:** Ensure the CRM system's integration methods are compatible with the systems it talks to. For example, if required and in case the core banking (Temenos Transact) and/or any other required systems which exposes SOAP web services, the proposed system should be able to consume those.

### 3.5 - Compliance & Regulatory

- e) **Regulatory Compliance:** State Bank requires that all customer complaints are resolved within a certain timeframe and reported. The system must facilitate compliance by capturing required data (like complaint received date, resolved date, reason for delay if any) and produce the required compliance reports. If the regulator has an interface (like a portal to

update complaint statuses or a required file format for monthly reporting), the system should help generate those outputs.

- f) **Internal Policy Compliance:** The CRM and its integrated CMS module must enforce internal SOPs (as covered in functional requirements).
- g) **Data Retention & Archiving:** The system could provide archiving functionality and must have the ability to export and purge old data in compliance with policies. Customer interaction data must be considered for long-term storage due to regulatory requirements and complaints records must be kept for 10 Years at least.
- h) **Disaster Recovery (DR):** Recovery Time Objective (RTO) and Recovery Point Objective (RPO) should meet Ubank's business needs as per defined SLA in this document.
- i) **User Training & Certification:** The bidder must ensure that the new system meets usability standards so that all agents can be certified on it. In this regard, the bidder must provide extensive and detailed training sessions along with training material.

## 4. Integration Points

The CRM system will act as a hub for various customer-related information flows. It must integrate seamlessly with multiple external systems and channels to provide unified experience. Key integration points include:

### 4.1 - IVR/Telephony System Integration

- a) The CMS will integrate with U Bank's call center telephony platform (and IVR).
- b) **Screen Pop & CTI:** As described earlier, when: When a call arrives, the telephony system will send a trigger (with caller ID, etc.) to the CMS. This can be achieved via a CTI (Computer Telephony Integration) link or API. CMS is required to use this data to automatically open the corresponding customer profile from the CRM or a new contact log screen for the agent, reducing manual lookup.
- c) **Data Dip:** The IVR must perform a data dip into the proposed CRM module, integrated core banking system (Temenos Transact) and any other system as per requirements, to authenticate or identify a customer before routing to an agent. For instance, a caller might input their account or CNIC in the IVR; the CRM can call an API to retrieve basic info from the core banking system and/or verify PIN attempts.
- d) **Call Control:** The CMS interface should have buttons to answer, hold, transfer, or terminate calls by utilizing Ubank's Call Center telephony system's API or a CTI middleware (like Cisco Finesse, Avaya CTI, etc.). The CMS should be flexible to integrate with standard CTI connectors.
- e) **Call Recording:** The telephony system records calls. The integration should enable the CMS to link to these recordings, possibly through an API or shared storage. The CMS ticket must have a "play recording" link that fetches the audio from the call recording system. Security controls are needed to ensure only authorized roles can play recordings.
- f) **Outbound Dialing:** CRM and integrated CMS is used for outbound calls (like follow-ups or lead calls), it should integrate to trigger calls via the dialer or phone system. For example, an agent clicks a customer's phone number in CRM or CMS component, and it sends a request to the phone system to place the call (click-to-dial functionality).
- g) **IVR Self-Service logs:** For transactions done on IVR (like balance inquiry, card block, Account block or digital channel block), the CRM must maintain a log by fetching a transcript or recording / message queue from via an API. CMS should also note that the customer had an IVR interaction prior to agent transfer.

### 4.2 - Core Banking System (CBS) Integration

U Bank's core branchless and branch banking systems (Temenos Transact and SAP) holds account and transaction data.

- a) **Customer Data Inquiry:** When a customer is identified in CRM, the system will fetch their details from CBS in real-time (or from a replicated customer database if available) via an API or stored procedure provided by CBS. Modern APIs or web services should be to retrieve the data which includes accounts, balances, card info as listed in 360 degree customer view.
- b) **Transaction History:** For showing recent transactions or balances, the CRM may call core banking's services to get mini-statements or account status. E.g., an API that returns last N transactions for a given account number.
- c) **Service Requests:** Some complaints are resolved by performing actions in CBS (like updating customer info, reversing a charge, unblocking a card). Ideally, the CRM could trigger some of these actions via integration:
  - If APIs exist ("relevant APIs from system have to be extended to integrate with Transact" regarding cards), then the CRM can call an API to perform the action (like block card).
  - If APIs do not exist, the CRM will simply log the ticket request and a back-office user will manually do it in core banking.
- d) **Customer Flagging:** If a very high priority issue occurs (like fraud), possibly the CRM might set a flag on the customer in core (like a note or a freeze).
- e) **Batch Updates:** The CRM should perform a periodic sync for certain reference data from core (like updated branch list, product list, or nightly fetch of any new accounts to update the its customer index). This may be done via scheduled jobs or data replication.

### 4.3 - Data Lake / Data Warehouse Integration

U Bank's data lake will be a repository for analysis of various data including CMS data.

- a) **Data Export:** The CMS should send data to the data lake on a regular basis. Possibly, a nightly ETL process will extract the day's tickets, interactions, and outcomes from the CMS database and load into the data lake (for Hadoop/BigData, in JSON or CSV format; for SQL warehouse, via ETL tool). Alternatively, a streaming integration can push each new ticket or update as it happens to the lake (via a messaging queue or Kafka, etc.).
- b) **NPS/CSAT Data:** Surveys are conducted and stored in CRM, should flow to the data lake to combine with other sources (like transactional data) for deeper analysis.
- c) **Analytics Feedback:** In some cases, the data lake or an analytics model might send information back to the CRM. For example, if an analytics model predicts a customer is likely to churn, that insight could be fed into the CRM to alert agents when the customer calls.
- d) **Reporting Tool Integration:** U Bank uses PowerBI and MicroStrategy. On top of the data lake for reports, the service provider must ensure that the CRM's data is properly structured for integration with these tools.

## 4.4 - Mobile Banking Apps, Public Website and IMB Portal Integration

This section refers to U Bank's Internet Mobile Banking (IMB) Portal, Public Website and Mobile Applications (Ubank Core and Upaisa):

- a) **Complaint/Request Submission:** The CMS must be integrated to the mentioned segments via APIs as they shall call a secure REST API on the CMS backend to create a new ticket. The API will take inputs like user ID (which maps to customer), category, description, etc. and return a reference number and status.
- b) **Status Inquiry:** Likewise, the app/portal allows customers to check the status of previously lodged complaints. The CMS should expose an API to fetch status and details (limited info) of a ticket when provided with the reference number (and some authentication to ensure the customer only sees their own cases).
- c) **Push Notifications:** Integration is required extend to sending push notifications to the mobile apps regarding updates (if the user has the app and is logged in). For example, "Your complaint has been resolved." The CMS can send such a notification via the app's notification service or via the core banking's existing notification system.

## 4.5 - Email, SMS Gateways & Omni Channel Social Media

- a) **Email:** The CMS should integrate with the corporate email server (Microsoft Exchange and/or any other SMTP server) fetch incoming mails from a support inbox via IMAP/POP, and send outbound mails via SMTP. For sending, the CMS trigger may trigger an email through an SMTP integration whenever an agent sends a reply or the system sends an automated notification.
- b) **SMS:** Integration with an SMS gateway (through HTTP API or SMPP) to allow the CRM and integrated CMS to send text messages to customers (for OTP, notifications, or feedback requests) and also enable two-way SMS (for cases where customer replies). The content and triggers for these communications should be configurable (e.g., which events generate SMS). Also, the system should manage failures (e.g., if SMS fails to send, log it and possibly email instead) and perform retries.
- c) **Omni-channel Social Media Integration:** CRM incorporate integrations with social media channels (LinkedIn, Twitter, Facebook & Instagram). If a customer Tweets about an issue with U Bank, the CMS could automatically create a ticket or at least capture that mention for the social media team to respond.
- d) **WhatsApp and Messaging Apps Support:** The system must integrate with WhatsApp to allow customers to chat with U Bank support team via WhatsApp for opening tickets.
- e) **Customer 360 Enhancements with Credit Bureau and/or Tasdeeq:** The CRM must integrate data sources from Credit Bureau and/or Tasdeeq into the 360 Degree Customer view, to display if the customer's credit score details.

## 4.6 - External Complaint Portals

Interfacing with external complaint systems is required, such as the Prime Minister's Citizen Portal or State Bank's complaint portal i.e. SBP's Sunwai and RAAST (if customers log complaints there that need to be addressed by U Bank).

If, for example, a complaint is lodged on the Citizen Portal, U Bank's team receives it through a web interface or email. Ideally, the CMS could integrate by either (a) receiving an email that auto-creates a ticket, or (b) if an API or data feed is provided by that portal, pulling those complaints into CMS.

The Customer Interaction or category should indicate it came from an external channel (and maybe treat it with special priority). Once resolved, the CMS user would update the portal separately unless an API allows updating status back. This integration may be semi-manual if no direct interface exists, but the CMS should support tagging and reporting on such complaints (e.g., Customer Interaction for "Citizen Portal" complaint).

## 5. Licensing Requirements

The system should be used by high no. of users, therefore the system performance should be top-notch in case of high no. of concurrent users or the nature of activities that they may be performing, such as (but not limited to) lead generation, customer data queries, simultaneous call logging, searches, and updates etc. The solution must be highly scalable in the future to cater for any amount of workload.

**The tentative no. of users for license sizing:**

	<b>Year 1</b> (Initial Licenses)	<b>Year 2</b>	<b>Year 3</b>	<b>Year 4</b>	<b>Year 5</b>
<b>No. of Total Users</b>	800	880	968	1065	1172

**Note:** Ubank reserves the right to revise (scale up or scale down) the licensing requirements mentioned as per its discretion.

## 6. General Technology and Information Security Requirements

### 1. Information and Cyber Security Requirements:

- The solution must comply with all financial industry standard security requirements.
- Solution must adhere to Ubank's Information Security standards, attached in **Annexure-B: Compliance Sheet**, named **"Information Security"** to this RFP document.



- c. Solution must adhere to any other security guidelines by State Bank of Pakistan and / or Ubank's Information Security department, that are made public and intimated by Ubank before the sign-off of design phase.
- 2. All components / layers must be designed on an easy-to-execute horizontal and vertical scaling.

### 3. Implementation Methodology

- a. Vendor/Bidder must provide detailed implementation methodology, with tangible and measurable criteria for sign-off of each implementation phase.
- b. Post 1<sup>st</sup> cutover, vendor must propose an agile approach along with establishing a rapid Continuous Integration – Continuous Development (CICD) pipeline.

### 4. General Implementation Principles

- a. Vendor/Bidder must complete the standard test phases (unit testing, integration testing, user acceptance testing, performance testing, vulnerability testing and full-dress testing).
- b. All test cases must be provided on Jira, along with the results. No word or PDF documents for test cases will be acceptable.
- c. **Unit Testing:** Vendor/Bidder must provide detailed unit testing results for all functionality, along with proper screenshots and evidence, on JIRA.
- d. **Integration Testing:** Vendor/Bidder must provide detailed integration testing results to validate the integrations. In case no test environment for a third party is available or access to third party is not possible, vendor must develop stubs / mocks for bi-directional calls. Detailed test cases for integration testing and results of integration testing along with evidence (logs, screenshots, videos, etc.) must be uploaded by the vendor on Ubank's JIRA instance.
- e. **User Acceptance Testing:** Before UAT kickoff, Ubank will provide detailed list of UAT cases on JIRA. Vendor/Bidder must execute all UAT cases before formally calling out UAT. UAT results along with evidence (logs, screenshots, videos, etc.) must be uploaded by vendor on JIRA, before formal UAT by Ubank can be started.
- f. **Performance Testing:** Vendor/Bidder must execute at least 1 cycle of performance testing before production cutover. Detailed performance test cases, test results and summary outcome of performance testing cycle(s) will be provided by the vendor on JIRA. Performance testing results must comply with load forecast provided by Ubank as part of this RFP.
- g. **Vulnerability Assessment / Penetration Testing (VA/PT):** Vendor/Bidder must execute VA/PT cycles on the solution and all integrated layers, using standard VA/PT toolset, to be arranged by the Vendor. Ubank will also execute VA/PT cycles on all components using its toolset. Vendor/Bidder is responsible to close all critical, high, medium and low anomalies / vulnerabilities identified in any component of the solution, before going live.

5. **Training:** Vendor/Bidder must provide detailed training and transition plan (along with training delivery) at the start of the project, for transitioning implementation artifacts to Ubank's relevant teams and configuration / customization training of Ubank's resources. The training and transition plan should also ensure Ubank's resources are equipped with necessary skills to undertake Operations of the production platform from day 0.
6. **Hyper Care Period:**
  - a. Vendor /Bidder will provide **at least 3 months of Hypercare period post Go-Live.**
  - b. In Hypercare period, vendor will be responsible for managed operations as well as incident management, all under Ubank's team's supervision.
  - c. Vendor must agree to Hyper Care period SLA, with penalties in case of breach of SLA.
  - d. Hyper Care period will start from signoff of full 1<sup>st</sup> production deployment by Ubank.
7. **Warranty and Support:**
  - a. Vendor/Bidder must provide **1 year of warranty and support**, without any additional cost, for all components (may include but not limited to: software, license, services etc.) delivered as part of the solution.
  - b. The warranty period will require SLA based incident resolution; penalties would be applicable in case of SLA breach.
  - c. Warranty period will start at sign-off of Hyper Care period by Ubank.
  - d. Vendor/Bidder must also commit to providing **at least 4 years of support post warranty period**, with penalties backed SLA, as part of the proposal.
8. **Documentation:**
  - a. Vendor/Bidder must provide detailed documentation on the architecture and implementation of the solution.
  - b. Documentation must contain details of the architecture, including logical and physical architecture, deployment architecture and integration architecture.
  - c. The solution documentation must be provided in the form of a wiki / live document, instead of static word / pdf documents. Vendor/Bidder may choose to provide documentation in Jira Confluence or any other similar tool.
9. **Data Migrations:** Vendor/Bidder to perform historical data migrations from incumbent systems.

## 7. Service Level Agreement (SLA)

The proposed solution must be backed by a robust Service Level Agreement (SLA) covering, but not limited to, the following aspects:

1. **Availability:** Guaranteed uptime percentage of **99.99% (or higher)**, excluding scheduled maintenance.
2. **Incident Response, Restoration & Resolution Times:** The vendor must commit to the following maximum timelines for SLA:

Incident Severity	Coverage	Response Time	Restoration Time	Resolution Time	Target Thresholds
Severity 1	24 x 7 (including public holidays)	< 30 minutes	< 4 hours	< 2 calendar days	99%
Severity 2	24 x 7 (including public holidays)	< 1 hour	< 8 hours	< 4 calendar days	97%
Severity 3	Workdays	< 1 workday	< 3 workdays	< 6 workdays	95%
Severity 4	Workdays	< 2 workdays	< 5 workdays	< 10 workdays	90%
Technical Query	Workdays	< 24 hours	Best possible effort	Not Applicable	

Table 1: SLA for Incident Support

Severity Level	Description
Severity 1 – Critical Issue	Essential Business Process or Workgroup affected – any highly critical system or service outage that result in loss of defined business processes and/or capabilities, and for which there is no workaround acceptable to Ubank (availability of workaround renders it “High”)
Severity 2 – High Issue	Part of an Essential Business Process or Workgroup affected - Degradation of system or service performance that impacts end user service quality or significantly impairs business process control or operational effectiveness, but for which there is a workaround acceptable to Ubank.
Severity 3 – Low Issue	Part of essential/Non-Essential Business Process or Workgroup or Individual affected and can be replicated with discrete steps – Minor degradation of system or service performance that does not have any impact on end user service quality.
Severity 4 – Other Issue	A low-level priority defect that cannot be replicated, is caused by corruption in data, or has other non-software related cause
Technical Query	A query regarding technical information regarding a feature implemented

Table 2: Severity Level Definitions

Time Span	Description
Response Time	Response time is defined as the elapsed time between the vendor receiving the support request and formal acknowledgement of the request by the vendor back to the Ubank.
Restoration Time	Restoration time is defined as the elapsed time between response time and successful restoration of the service, by temporary or permanent fix or workaround, as confirmed and agreed by Ubank.
Resolution Time	Resolution time is defined as the elapsed time between restoration time and successful deployment of a permanent fix for the reported incident.

Table 3: Time Span Definition

3. **Performance:** Bidder to define & share metrics for system response times and UI responsiveness, report generation etc.
4. **Disaster Recovery:** In case of on-premises solution, the bidder must supply, deploy and configure a full-fledged Disaster Recovery (DR) site at U Bank Disaster Recovery DC to ensure seamless services, SLA compliance and automatic services shifting to DR in case any outage at primary site.
5. **Support Channels:** Availability of 24/7 support via multiple channels (phone, email, chat, dedicated portal).
6. **Data Backup & Recovery:** Bidder to share details on backup frequency, retention policies, and disaster recovery plans with defined Recovery Point Objective (RPO) and Recovery Time Objective (RTO).
7. **Security Incident Response:** Bidder to define & share the process and timelines for responding to and notifying Ubank of any security breaches or incidents.
8. **Penalties:** Clear penalties for failure to meet agreed-upon SLA targets.
9. **Reporting:** Regular performance reports on SLA adherence.
10. **Escalation Matrix:** The vendor must provide a clear escalation matrix for all issues, incidents, and change requests related to their provided solution. This matrix should outline the contact points and escalation paths within the vendor's organization.

Escalation Level	Role/Contact Person	Contact (Email/Phone)	Method	Response Expectation	Time
Level 1	Service Desk / L1 Support				
Level 2	Team Lead / L2 Support				
Level 3	Manager / L3 Support				
Level 4	Director / Executive				

## 11. Service and Website performance metrics:

Metric Category	Parameter	Target Threshold	Definition
Core User Experience	Largest Contentful Paint (LCP)	< 1.0 seconds	<b>Measures loading performance.</b> This is the time it takes for the largest image or text block to become visible within the viewport.
	Interaction to Next Paint (INP)	< 200 milliseconds	<b>Measures interactivity.</b> This metric assesses the delay from when a user interacts with the page (e.g., clicks a button) until the page visually responds. It has largely replaced First Input Delay (FID).
	Cumulative Layout Shift (CLS)	< 0.1	Measures <b>visual stability</b> . This checks how much the content unexpectedly shifts or "jumps around" during loading. A low score is better.
Server & Network	Time to First Byte (TTFB)	< 500 milliseconds	This <b>measures how long it takes for a user's browser to receive the very first byte of data from the server</b> after making a request. It's a key indicator of server responsiveness.
	Uptime / Availability	> 99.99%	This is the <b>percentage of time the service is online and operational</b> . This is a critical Service Level Agreement (SLA) metric.
	Error Rate	Less than <b>0.1%</b> of all server requests.	The <b>percentage of server requests that result in an error</b> (e.g., HTTP 5xx server errors).

## 8. Glossary

**360-Degree View (360° View):** A comprehensive view of the customer’s information in one place, including personal details, all products and accounts they hold, and their history of interactions with the bank. In this document, it refers to the CMS screen that aggregates data from various systems to give a full picture of the customer.

**AI (Artificial Intelligence):** Broadly, the simulation of human intelligence processes by machines. In this context, it refers to features like machine learning models for routing, natural language processing for sentiment and chatbots, etc.

**API (Application Programming Interface):** A set of protocols and tools for building software and allowing different systems to communicate. Mentioned often for integrations (e.g., CRM calling core banking API).

**Audit Trail:** A chronological record of all changes or actions performed in the system, used for tracking who did what and when. Essential for compliance and debugging.

**Chatbot:** An AI-driven chat program that can converse with users in natural language to answer questions or perform tasks. Integrated into the CMS to handle routine queries in text form.

**Core Banking System (CBS):** The primary system of record for banking transactions (accounts, balances, etc.). For U Bank’s Branch Banking customers, this is Temenos Transact and for U Bank’s Branchless Banking customers (i.e., Upaisa), this is SAP. The CRM connects to it to fetch or update customer financial data.

**CRU (Complaint Resolution Unit):** A department or team focused on resolving customer complaints and queries, usually back-office specialists beyond the front-line call agents.

**CSR (Customer Service Representative):** In this context, a call center agent or branch staff who serve customers. Also often just called “agent.”

**CSAT (Customer Satisfaction Score):** A metric typically collected via surveys asking customers to rate their satisfaction with the service (often on a scale). Used to gauge short-term satisfaction.

**Data Lake:** A centralized repository for storing large volumes of raw data in native format. U Bank uses it to aggregate data (like complaints data) for analytics and reporting.

**ETB / NTB:** Existing-to-Bank / New-to-Bank. ETB refers to an existing customer (with records in the system), NTB is a new customer (no prior records). Used in processes to handle verification differently.

**IVR (Interactive Voice Response):** The automated phone menu system that callers interact with (e.g., “Press 1 for English, 2 for Urdu...”). It can handle some queries via automation or route to live agents.

**Knowledge Base (KB):** A repository of help articles, FAQs, and procedural guides that agents or customers can refer to for information and troubleshooting.

**IMB:** Internet Mobile Banking portal for Ubank’s branch banking customers.

**Mobile App:** Refers to the mobile banking app for Ubank’s Branch Banking customers and Upaisa’s Branchless Banking Customers.

**KPI (Key Performance Indicator):** A measurable value that indicates how effectively a company is achieving key objectives. Here, KPIs could be average resolution time, SLA compliance, NPS, etc., for the customer service operations.

**Main Type / Sub-Type:** Refers to the hierarchical categorization in Work Codes. Main Type is a broad category of issue, Sub-Type (or Sub work code) is a specific issue under that category.

**NLP (Natural Language Processing):** A branch of AI that deals with understanding and generating human language. Used in sentiment analysis, chatbots, transcription analysis, etc.

**NPS (Net Promoter Score):** A customer loyalty and satisfaction measurement obtained by asking customers how likely they are to recommend the company to others (scale of 0-10). Calculated as %Promoters minus %Detractors. It's a key metric for overall customer experience.

**OCR (Optical Character Recognition):** Not explicitly mentioned, but if attachments need text reading. In general, OCR converts images of text into machine-readable text.

**Predictive Routing:** Use of AI to decide where a ticket should go by predicting outcomes based on historical data (an AI-enhanced form of skills-based routing).

**Priority (in tickets):** A flag (such as Low, Medium, High, Critical) indicating the urgency and importance of a ticket, which can affect how quickly it should be handled. Priority can be set manually or by rules/AI.

**RPA (Robotic Process Automation):** Tools that automate repetitive tasks by mimicking user actions. Mentioned as a future enhancement to automate tasks in other systems triggered by CMS events.

**SLA (Service Level Agreement):** The target time within which a service (like resolving a complaint) should be delivered. Often defined per category of issue. E.g., 3 days for minor issues, 1 day for urgent. Used to measure performance and set customer expectations.

**Speech Analytics:** The process of analyzing recorded calls to gather information (transcription, sentiment, keywords, etc.). In CMS, used for quality monitoring and sentiment.

**TAT (Turnaround Time):** Similar to SLA, the actual time it takes to resolve a request/complaint. SOPs often define TAT for each Customer Interaction (e.g., 6 days, 3 days as in the document snippets).

**Ticket:** A generic term for a logged case in the CMS (could be a complaint, service request, inquiry, etc.). It's the entity that tracks an issue from report to resolution.

**UAT (User Acceptance Testing):** The phase where actual users test the system in a real-world scenario to ensure it meets their needs before it is officially accepted and goes live.

**VIP Customer:** A very important customer (could be high net worth or key relationship). Often these are handled with special care (shorter SLAs, personal follow-ups).

**Work Code:** In U Bank's CMS context, a coded classification for the type of customer contact or complaint. It usually corresponds to an issue category and drives workflows. E.g., code 125 Remittance Issue, subcode 1167 Funds transfer issue. This term appears frequently in SOPs and requirements.

## END OF DOCUMENT