VPBank

powered by aws

VPBANK TECHNOLOGY HACKATHON 2025

Challenge: Speak-to-Input

Technical Documentation

VPBank Speak-to-Input
Multi Agent Automation System

Team 19: PipeKat and LodiKat

1. Pham Nguyen Hai Anh (Leader)

2. Bui Ho Ngoc Han

3. Danh Hoang Hieu Nghi

4. Le Minh Nghia

5. Nguyen Duc Toan

November, 2025

Table of Contents

1. Executive Summary 6
1.1 Project Overview 6
1.2 Business Objectives 6
1.3 Key Business Value 6
1.4 Solution Scope 6
2. Problem Definition 7
2.1 Current State Challenges 7
2.1.1 Loan Origination System (LOS) Bottlenecks 7
2.1.2 Human Resource Management System (HRMS) Inefficiencies 7
2.1.3 Customer Relationship Management (CRM) Gaps 8
2.1.4 Risk & Compliance Entry Challenges 8
2.2 Quantified Business Impact 8
2.3 Root Causes 8
3. Objectives & Key Performance Indicators (KPIs) 10
3.1 Primary Objectives 10
3.1.1 Operational Efficiency 10
3.1.2 Data Quality & Accuracy 10
3.1.3 Customer Experience Enhancement 10
3.1.4 Cost Reduction 10
3.2 Key Performance Indicators (KPIs) 10
3.3 Success Criteria 11
4. System Overview 12
4.1 Architecture Principles 12
4.1.1 Simplicity 12
4.1.2 Scalability 12
4.1.3 Security 12
4.1.4 Cost Efficiency 12
4.2 Core Technology Stack 13
4.3 User Flow Overview 13
5. Architecture & Infrastructure 15
5.1 Logical Architecture 15
5.1.1 Presentation Tier 15
5.1.2 Application Tier 15
5.1.3 AI Services Tier 15
5.1.4 Data & Integration Tier 15
5.2 AWS Deployment Architecture 16
5.2.1 Network Layer 16
5.2.2 Compute Layer 16

5.2.3 Security Layer....17
5.2.4 Data Layer....17
5.2.5 Monitoring & Observability....17
6. Data Flow Diagrams (DFD Levels 0-3)....18
6.1 Level 0 Data Flow Diagram....18
6.2 Level 1 Data Flow Diagram....18
Process 1: Process Voice....19
Process 2: Act on Intent....19
6.3 Level 2 Data Flow Diagram....20
Process 1.1: Cancel Noise....20
Process 1.2: Detect Voice Activity....20
Process 1.3: Detect Turn....20
Process 1.4: Transcribe....21
Process 2.1: Recognize Intent....21
Process 2.2: Action on Intent Banking Operations....21
Process 2.3: Interact Banking UI....21
Process 2.4: Validate & Confirm....21
6.4 Level 3 Data Flow Diagrams - Use Cases....21
6.4.1 Loan Submission Workflow....22
6.4.2 CRM Customer Update Workflow....22
6.4.3 HR Job Posting Workflow....23
6.4.4 Compliance Validation Workflow....24
7. Component Design....25
7.1 Voice Pipeline Component....25
7.1.1 Component Overview....25
7.1.2 Core Responsibilities....25
7.2 Component Details – Client Layer....25
7.3 Component Details – Voice Processing....27
7.4 Component Details – AI Agent System....30
Compliance Agent....31
Loan Agent....33
CRM Agent....33
HR Agent....33
Browser–Use Execution Agent....34
7.5 Speech Recognition Component (PhoWhisper)....35
7.5.1 Fine-Tuned Model Specifications....35
7.5.2 Technical Specifications....35
8. API & Schema Definition....36
8.1 API <1>....36

8.3 DynamoDB Schema Definitions 36
9. Evaluation & Testing 37
9.1 Evaluation Metrics 37
9.2 Testing Strategy 37
9.2.1 Unit Testing 37
9.2.2 Integration Testing 37
9.2.3 User Acceptance Testing (UAT) 37
10. Cost & ROI Analysis 38
10.1 AWS Service Cost Breakdown (for prototype) 38
10.2 Production Scale Cost Estimation 39
10.2.1 Scaling Assumptions 39
10.2.2 AWS Service Cost Breakdown (Production Scale) 39
10.3 Implementation Costs 40
10.4 Operational Cost Savings 40
Baseline Manual Processing Costs 40
With AI Automation (Production Scale) 40
10.5 ROI Calculation 41
Monthly & Annual Savings 41
First Year Financial Summary 41
ROI Metrics 41
10.7 Additional Business Value (Intangible Benefits) 41
11. Security & Compliance 43
11.1 Security Architecture 43
11.1.1 Authentication & Authorization 43
11.1.2 Data Encryption 43
11.2 Network Security 43
11.3 Audit 43
12. Deployment Guide 44
12.1 Prerequisites 44
12.2 Deployment Architecture 44
12.3 Infrastructure Deployment 45
12.3.1 Terraform Configuration 45
12.3.2 Key Infrastructure Components Deployed 45
12.4 Application Deployment 48
12.4.1 Container Image Build & Push 48
12.4.2 ECS Task Definition Update 49
12.5 CI/CD Pipeline 49
12.6 Configuration Management 49
12.7 Rollback Procedures 49

13. Monitoring & Maintenance 51
13.1 System Monitoring 51
13.1.1 CloudWatch Dashboards 51
13.1.2 Key Performance Indicators (KPIs) 51
13.2 Model Maintenance 51
13.2.1 PhoWhisper Model Fine-tuning 51
13.2.2 Intent Classification Model Updates 51
13.3 Database Maintenance 52
13.4 Log Management 52
13.5 Capacity Planning 52
14. Future Enhancements 53
14.1 Near-Term Enhancements (3-6 months) 53
14.2 Medium-Term Enhancements (6-12 months) 53
15. Project plan 54
Total Duration: 18 days (October 20 – November 6, 2025) 54
6. Team Structure (5 Members) 54
7. Daily Task Distribution 55
16. Appendices 55
16.1 Glossary of Terms 56
16.3 Sample API Requests & Responses 56
16.4 Test Scenarios & Scripts 56
16.5 Validation Results 56
15.6 References 57

1. Executive Summary

1.1 Project Overview

The VPBank Speak-to-Input Multi Agent Automation System is an advanced voice-driven solution designed to revolutionize data entry and workflow automation across banking operations. By leveraging state-of-the-art Generative AI technologies, fine-tuned Vietnamese speech recognition, and intelligent multi-agent orchestration, this system transforms manual, time-consuming data entry processes into seamless voice-activated workflows.

1.2 Business Objectives

- Reduce manual data entry time by 73%, from average 6 minutes 30 seconds to 1 minute 45 seconds per transaction [1].
- Decrease data entry error rates from 1-4% (manual) to below 0.5% (automated) through AI-powered validation [2][3].
- Automate up to 60% of identified repetitive back-office banking workflows including Loan Origination, CRM updates, HR processes, and Compliance reporting [4].
- Enhance customer experience through hands-free, voice-activated service delivery.
- Improve accessibility for elderly customers (11.6% of Vietnamese population aged 60+ as of 2019 census [6]) and persons with disabilities (7.06% [7]).
- Achieve operational cost reduction through AI automation, with estimated transaction cost of $0.002-0.015 per optimized AI-assisted operation [5].

1.3 Key Business Value

This solution delivers transformative value across multiple dimensions: operational efficiency through automation of high-volume repetitive tasks, enhanced accuracy via AI-powered validation reducing costly errors, improved customer experience with voice-first interfaces, increased accessibility for underserved populations, and significant cost savings with AI operations costing a fraction of manual processing.

1.4 Solution Scope

The system encompasses four primary use cases across VPBank's operations:

1. Loan Origination & KYC Automation: Voice-assisted loan application processing with automated form completion and document verification.

2. CRM Update & Customer Interaction: Real-time customer information management through voice commands.

3. HR & Internal Workflow Automation: Voice-driven job postings, leave requests, and employee record management.

4. Compliance Validation & Reporting: Automated compliance checks and regulatory report generation.

Metric	Current State	Target State
Average Processing Time	6m 30s per transaction	1m 45s per transaction (73% faster)
Data Entry Error Rate	1-4% manual error rate [2]	<0.5% with AI validation
Form Completion Rate	42-58% (industry average) [8]	**75-85% with voice interface**
Cost per Transaction	$5-10 (manual processing)	$0.002-0.015 (optimized AI automation)

2. Problem Definition

2.1 Current State Challenges

Vietnamese banking operations face significant productivity and quality challenges stemming from heavy reliance on manual data entry across core banking systems. These inefficiencies directly impact operational costs, customer satisfaction, and competitive positioning.

2.1.1 Loan Origination System (LOS) Bottlenecks

- Manual re-entry of customer and financial data across multiple system modules during credit evaluation
- Processing delays leading to extended loan approval cycles (average 3-5 business days)
- Data inconsistencies between application forms and internal systems requiring rework
- High abandonment rates due to lengthy application processes

2.1.2 Human Resource Management System (HRMS) Inefficiencies

- Manual updates for employee records, payroll adjustments, and attendance tracking
- Data redundancy across multiple HR systems creating synchronization issues
- Increased error rates in payroll processing (1-4% manual entry error rate)

- Delayed workforce reporting hindering strategic HR decisions

2.1.3 Customer Relationship Management (CRM) Gaps

- Manual entry of customer interactions and service requests leading to fragmented records
- Reduced data accuracy impacting customer service quality
- Slower response times limiting personalized customer engagement
- Difficulty in maintaining complete customer interaction history

2.1.4 Risk & Compliance Entry Challenges

- Compliance officers in Vietnamese banks spend excessive time manually entering multi-field risk assessment data across numerous regulatory reporting forms, leading to delayed submission of critical compliance reports to the State Bank of Vietnam.
- Manual data entry of customer due diligence information and transaction monitoring details is highly susceptible to typographical errors, potentially causing false positives in anti-money laundering (AML) detection systems and compliance violations.
- Vietnamese-specific compliance requirements involve complex data fields with specialized terminology that are difficult to input accurately, especially when staff must switch between Vietnamese and English technical terms.
- The high volume of manual compliance data entry creates bottlenecks during peak audit periods and regulatory inspections, preventing risk officers from focusing on higher-value analytical tasks that require human judgment and expertise.

2.2 Quantified Business Impact

Impact Area	Quantified Impact
Operational Productivity Loss	500 staff hours monthly on repetitive data entry tasks
Error-Related Costs	1-4% error rate [2] causing rework, compliance penalties, and customer dissatisfaction
Customer Experience Impact	67% of customers [8] abandon lengthy forms; 42-58% average form completion [8] rate
Competitive Disadvantage	Slower turnaround times compared to digitally-advanced competitors

2.3 Root Causes

1. **Legacy System Limitations:** Core banking systems designed without modern automation capabilities

2. **Manual Process Dependencies:** Heavy reliance on keyboard-based data entry without alternative input methods

3. **Lack of Intelligent Automation:** Absence of AI-powered validation and workflow orchestration

4. **Multi-System Data Silos:** Disconnected systems requiring duplicate data entry

5. **Limited Accessibility Options:** No voice-based alternatives for customers with disabilities or elderly users

3. Objectives & Key Performance Indicators (KPIs)

3.1 Primary Objectives

3.1.1 Operational Efficiency

- Reduce average transaction processing time from 6m 30s to 1m 45s (73% improvement)
- Achieve 80% reduction in process time for loan origination, and account reconciliation
- Automate 60% of monthly transactions through voice-activated workflows

3.1.2 Data Quality & Accuracy

- Achieve Word Error Rate (WER) below 5% for Vietnamese speech recognition

$$WER = (Substitutions + Deletions + Insertions) / Total\ Words \times 100\%$$

- Reduce data entry error rate from 1-4% to below 0.5% through AI validation

3.1.3 Customer Experience Enhancement

- Increase form completion rate from 42-58% to 75-85%
- Improve Customer Effort Score (CES) by enabling natural voice interactions
- Achieve Net Promoter Score (NPS) increase through enhanced service delivery

3.1.4 Cost Reduction

- Reduce cost per transaction from $5-10 to $0.01-0.02 through automation
- Achieve ROI within 12 months of full deployment
- Decrease operational costs by 10-30% through AI-powered process automation

3.2 Key Performance Indicators (KPIs)

KPI Category	Metric	Baseline	Target
Technical Accuracy	Word Error Rate (WER)	7-9% (standard)	**<5%**
Process Quality	Workflow Correction Rate	N/A	**>95%**
System Performance	End-to-End Voice Latency	N/A	**<4.0s**
User Experience	User Satisfaction Score	Baseline TBD	**>85%**
Business Impact	Form Completion Rate	42-58% [8]	**75-85%**
Cost Efficiency	Processing Time	6m 30s [1]	1m 45s (73% ↓)

3.3 Success Criteria

1. **Technical Success:** System achieves <5% WER for Vietnamese speech and >95% workflow accuracy within 1 month of deployment

2. **Operational Success:** 60% of target workflows automated with 73% reduction in processing time

3. **User Adoption:** 85% user satisfaction score and >75% form completion rate

4. **Financial Success:** Demonstrated ROI within 12 months (in production) through cost savings and efficiency gains

4. System Overview

The solution overview is designed in the figure below:

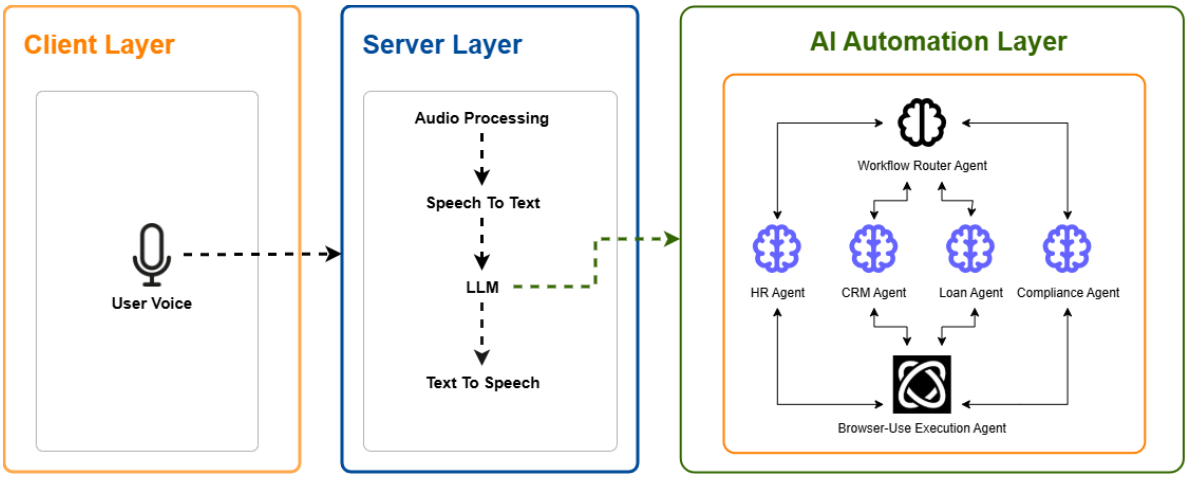


Figure: Abstract System Overview Diagram

4.1 Architecture Principles

4.1.1 Simplicity

The system follows a three-layer architecture pattern:

- **Client Layer:** User interaction interface with WebRTC-based real-time audio streaming
- **Server Layer:** Voice processing (STT/TTS) and AI automation orchestration
- **Integration Layer:** Domain-specific AI agents interfacing with banking systems

4.1.2 Scalability

Microservice-based architecture with:

- Amazon ECS with Fargate for containerized compute
- ECS Service Auto Scaling for dynamic capacity adjustment
- Independent scaling of voice processing and automation layers

4.1.3 Security

- API Gateway as centralized security entry point
- Amazon Cognito for RBAC integrated with VPBank SSO
- End-to-end encryption for WebRTC voice streams
- AES-256 encryption at rest (AWS KMS) and TLS 1.2+ in transit

4.1.4 Cost Efficiency

- Serverless-first design leveraging managed AWS services
- Usage-based pricing for AI services (Amazon Bedrock, PhoWhisper, ElevenLabs)
- Cost optimization through auto-scaling and pay-per-use model

4.2 Core Technology Stack

Component	Technology & Purpose
Speech Recognition	Fine-tuned PhoWhisper: Vietnamese-optimized ASR with <5% WER, handling regional accents and banking terminology
AI Orchestration	Amazon Bedrock (Claude Sonnet): Intent recognition, data extraction, validation, and workflow routing
Multi-Agent Framework	LangChain/LangGraph: Coordinating specialized agents (Loan, CRM, HR, Compliance) for domain-specific tasks
Browser Automation	Browser-Use Agent: Automated form filling, button clicking, and web interface interaction
Voice Pipeline	Pipecat: Real-time audio streaming, VAD, turn detection, and WebRTC orchestration
Text-to-Speech	ElevenLabs: Natural Vietnamese voice synthesis for interactive confirmation and feedback
Voice Enhancement	Silero VAD for noise cancellation and speech activity detection
Cloud Infrastructure	AWS ECS, Fargate, DynamoDB, S3, CloudWatch, API Gateway, Cognito, KMS

4.3 User Flow Overview

The typical user interaction follows a streamlined voice-to-action flow:

6. **Voice Capture:** User speaks command through browser/mobile interface with WebRTC audio capture
7. **Audio Enhancement:** Silero VAD for noise cancellation and active speech segments detection
8. **Speech Recognition:** Fine-tuned PhoWhisper transcribes Vietnamese speech to text
9. **Intent Recognition:** Coordinator AI analyzes text and extracts intent, entities, and required actions
10. **Agent Routing:** Workflow Router delegates task to specialized agent (Loan, CRM, HR, or Compliance)

11. **Action Execution:** Browser-Use agent automates form filling and button interactions on banking interfaces
12. **Validation & Confirmation:** System validates results and provides voice/text feedback via ElevenLabs TTS
13. **Completion:** Transaction logged for audit trail with full traceability

5. Architecture & Infrastructure

5.1 Logical Architecture

The VPBank Speak-to-Input system implements a modern, cloud-native architecture designed for scalability, security, and maintainability. The logical architecture is organized into four primary tiers:

5.1.1 Presentation Tier

- **Web Interface:** React-based responsive UI delivered via CloudFront CDN and S3 static hosting
- **WebRTC Client:** Browser-based real-time communication for low-latency voice transmission

5.1.2 Application Tier

- **API Gateway:** Centralized request routing, authentication, rate limiting, and logging
- **Pipecat Voice Engine:** Containerized on ECS Fargate, orchestrates audio streaming and processing pipeline
- **Multi-Agent Orchestrator:** LangGraph-based coordination of domain-specific AI agents
- **Browser Automation Service:** Headless browser instances for form manipulation and UI interaction

5.1.3 AI Services Tier

- **Speech-to-Text:** Fine-tuned PhoWhisper models deployed on SageMaker for Vietnamese transcription
- **Language Understanding:** Amazon Bedrock Claude Sonnet for intent recognition and entity extraction
- **Text-to-Speech:** ElevenLabs API for natural Vietnamese voice synthesis
- **Voice Enhancement:** Silero VAD for noise cancellation and audio quality optimization

5.1.4 Data & Integration Tier

- **State Management:** DynamoDB for conversation state, session data, and audit logs
- **Document Storage:** S3 for voice recordings, transcriptions, and system artifacts
- **Knowledge Base:** OpenSearch for regulatory documents, policies, and RAG retrieval

- **Banking System Integration:** REST APIs and event-driven connectors to LOS, CRM, HRMS, and Compliance systems

5.2 AWS Deployment Architecture

The system is deployed on AWS using Infrastructure as Code (Terraform) with the following key components:

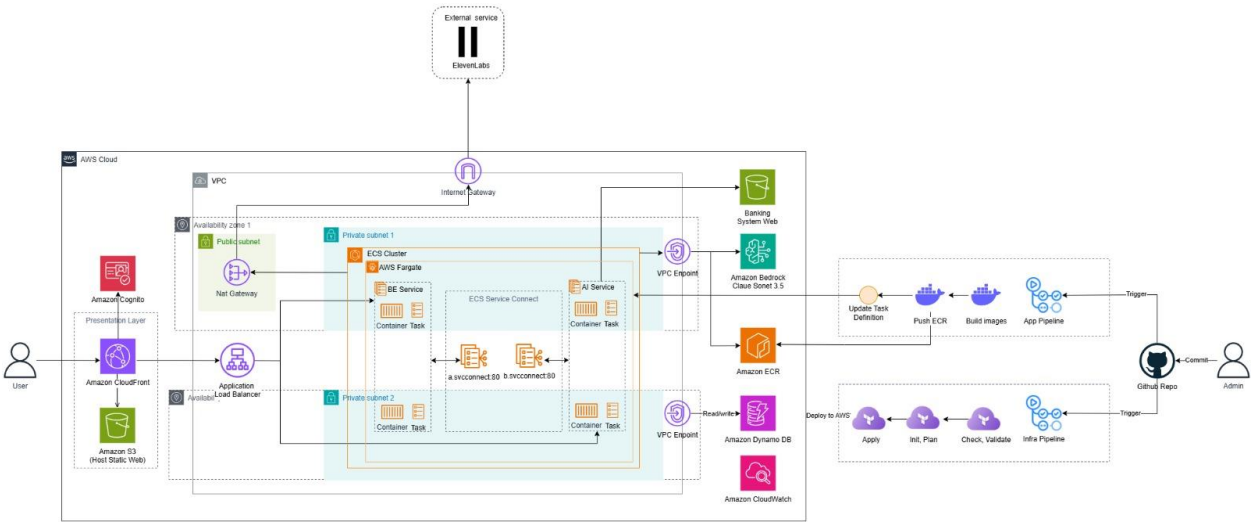


Figure <number>: AWS Deployment Diagram

5.2.1 Network Layer

- **VPC Configuration:** Multi-AZ deployment across 2 availability zones for high availability
- **Private Subnets:** ECS Fargate tasks run in private subnets without public IP exposure
- **NAT Gateway:** Outbound internet access for external API calls (ElevenLabs, model updates)
- **VPC Endpoints:** Private connections to Bedrock, DynamoDB, S3, ECR without internet egress

5.2.2 Compute Layer

- **ECS Fargate Services:** Serverless container execution for Pipecat engine and agent orchestrator
- **Service Auto Scaling:** CPU and memory-based scaling policies for dynamic load adjustment
- **Container Registry:** Amazon ECR for Docker image storage with vulnerability scanning

- **Service Connect:** Service discovery and communication between Pipecat and agent services

5.2.3 Security Layer

- **Amazon Cognito:** User authentication and RBAC integrated with VPBank SSO
- **AWS KMS:** Encryption key management for data at rest
- **Secrets Manager:** Secure storage for API keys, database credentials, and service tokens
- **Security Groups:** Firewall rules restricting traffic to necessary ports and protocols
- **IAM Roles:** Least-privilege access policies for service-to-service communication

5.2.4 Data Layer

- **DynamoDB Tables:** Conversation state, user sessions, workflow tracking, audit logs
- **S3 Buckets:** Voice recordings, transcriptions, model artifacts, compliance documents
- **OpenSearch:** Full-text search for knowledge base and policy documents (RAG)

5.2.5 Monitoring & Observability

- **CloudWatch Logs:** Centralized logging for all services with 7-year retention
- **CloudWatch Metrics:** Real-time performance metrics (latency, throughput, error rates)
- **CloudWatch Alarms:** Automated alerting for critical thresholds and anomalies
- **CloudTrail:** API activity logging for security and compliance auditing

6. Data Flow Diagrams (DFD Levels 0-3)

6.1 Level 0 Data Flow Diagram

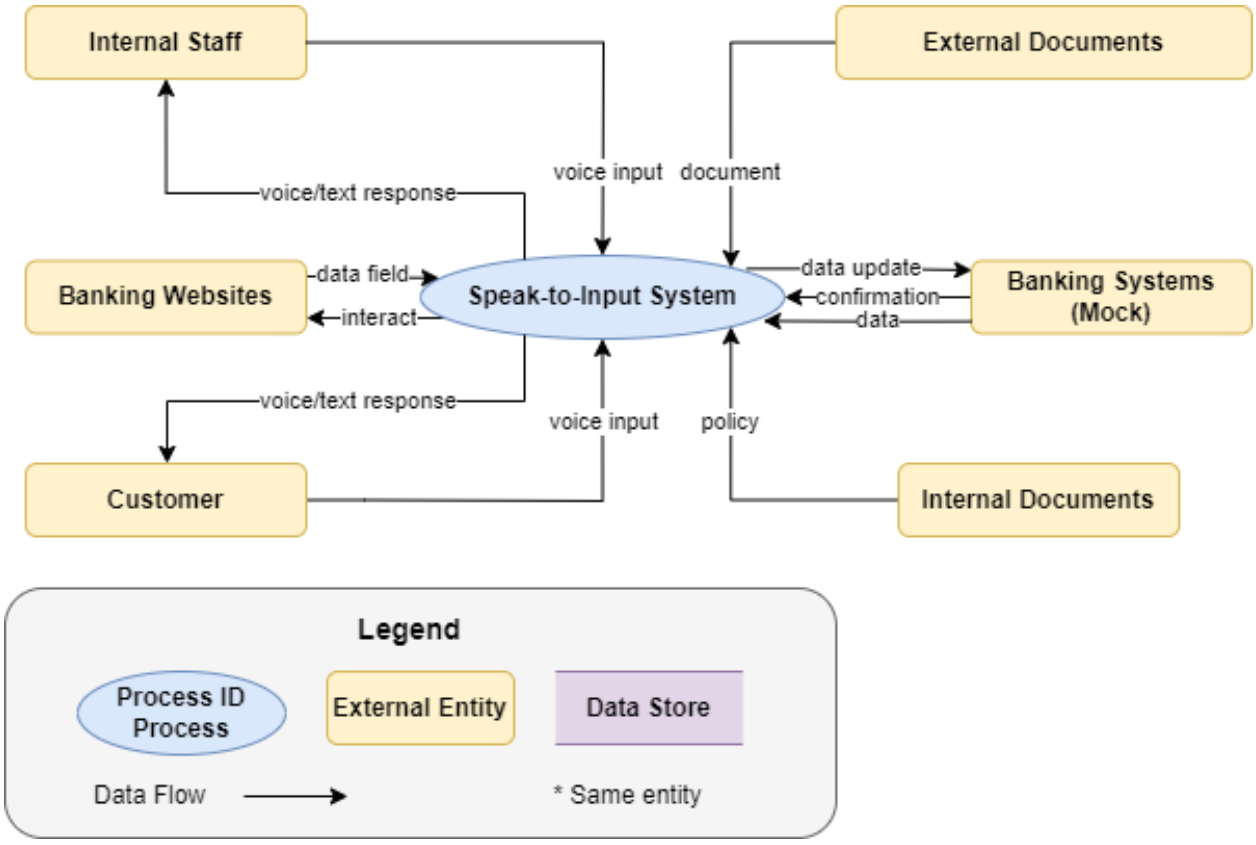


Figure 1: Level 0 Data Flow Diagram

External Entities:

- User (Customer or Staff member)
- Banking Systems (LOS, CRM, HRMS, Compliance Portal)
- External Document Sources (Regulatory docs, internal policies)

Process Flow:

14. User provides voice input to the Speak-to-Input System
15. System retrieves relevant context data from External Document Sources
16. System executes actions on Banking Site based on voice commands
17. System provides voice or text feedback to User confirming completion

6.2 Level 1 Data Flow Diagram

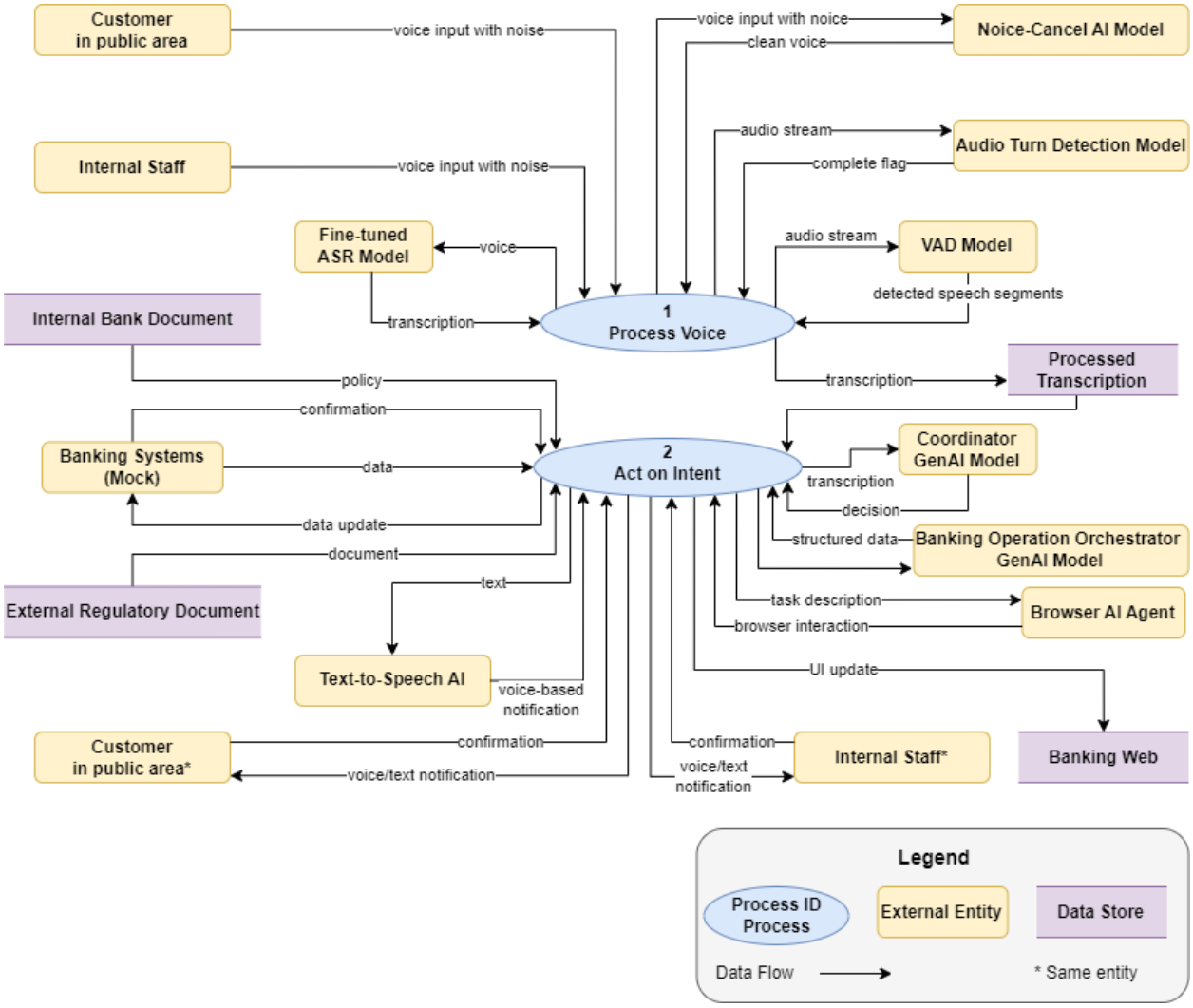


Figure 2: Level 1 Data Flow Diagram

Level 1 decomposes the main system into two primary processes:

Process 1: Process Voice

Transforms raw voice input into clean, structured text through multiple sub-processes:

- **Noise Cancellation (Silero VAD):** Filters background noise and audio artifacts
- **Voice Activity Detection (Silero VAD):** Identifies speech segments and filters silence
- **Turn Detection (Smart-Turn):** Detects conversation turns for multi-speaker interaction
- **Transcription (PhoWhisper):** Converts Vietnamese speech to text with <5% WER

Process 2: Act on Intent

Interprets transcribed text and executes corresponding actions:

- **Intent Recognition (Coordinator GenAI):** Classifies user intent and extracts key entities

- **Agent Orchestration (Amazon Bedrock + LangGraph):** Routes to appropriate domain agent
- **Browser Automation:** Executes form filling, editing, submission on banking interfaces
- **Confirmation (TTS):** Provides voice/text feedback to user

6.3 Level 2 Data Flow Diagram

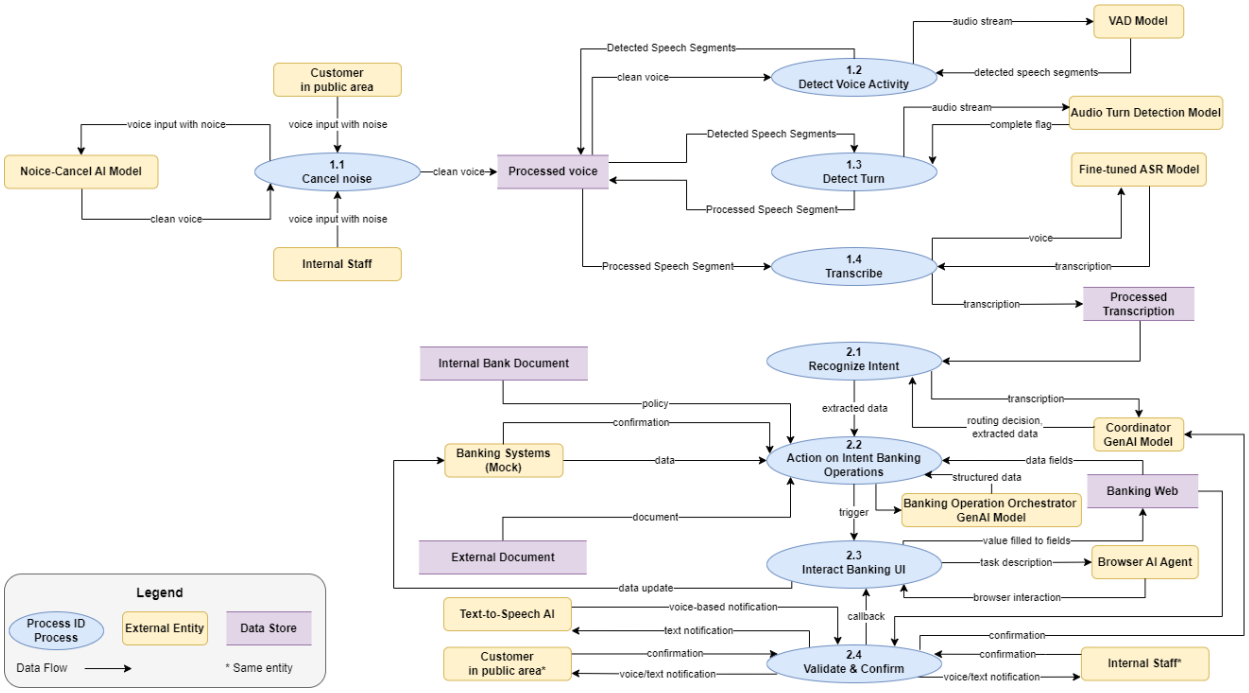


Figure 3: Level 2 Data Flow Diagram

Level 2 provides detailed decomposition of core processes:

Process 1.1: Cancel Noise

Input: Raw voice audio from user

Processing: VAD AI Model filters environmental noise

Output: Clean audio stream

Process 1.2: Detect Voice Activity

Input: Clean audio stream

Processing: VAD Model (Silero) detects speech vs. silence

Output: Speech segments for transcription

Process 1.3: Detect Turn

Input: Speech segments

Processing: Smart-Turn Model identifies speaker boundaries

Output: Segmented audio by speaker turn

Process 1.4: Transcribe

Input: Segmented speech audio

Processing: Fine-tuned PhoWhisper ASR converts speech to text

Output: Vietnamese text transcription

Process 2.1: Recognize Intent

Input: Text transcription

Processing: Coordinator GenAI extracts intent and entities

Output: Structured intent data (action type, entities, parameters)

Process 2.2: Action on Intent Banking Operations

Input: Structured intent data

Processing: Banking Operation Orchestrator routes to domain agent, validates with banking systems, retrieves policies

Output: Validated structured data ready for UI interaction

Process 2.3: Interact Banking UI

Input: Validated structured data

Processing: Browser AI Agent performs form filling, button clicks, data retrieval

Output: Success/failure confirmation

Process 2.4: Validate & Confirm

Input: Transaction results

Processing: System verifies completion, updates records, generates notifications

Output: Voice/text confirmation to user via TTS

6.4 Level 3 Data Flow Diagrams - Use Cases

Level 3 DFDs detail specific workflow implementations for each banking use case. The system supports four primary workflows: Loan Submission, CRM Customer Update, HR Job Posting, and Compliance Validation. Each workflow follows the same core process pattern while implementing domain-specific logic through specialized AI agents.

6.4.1 Loan Submission Workflow

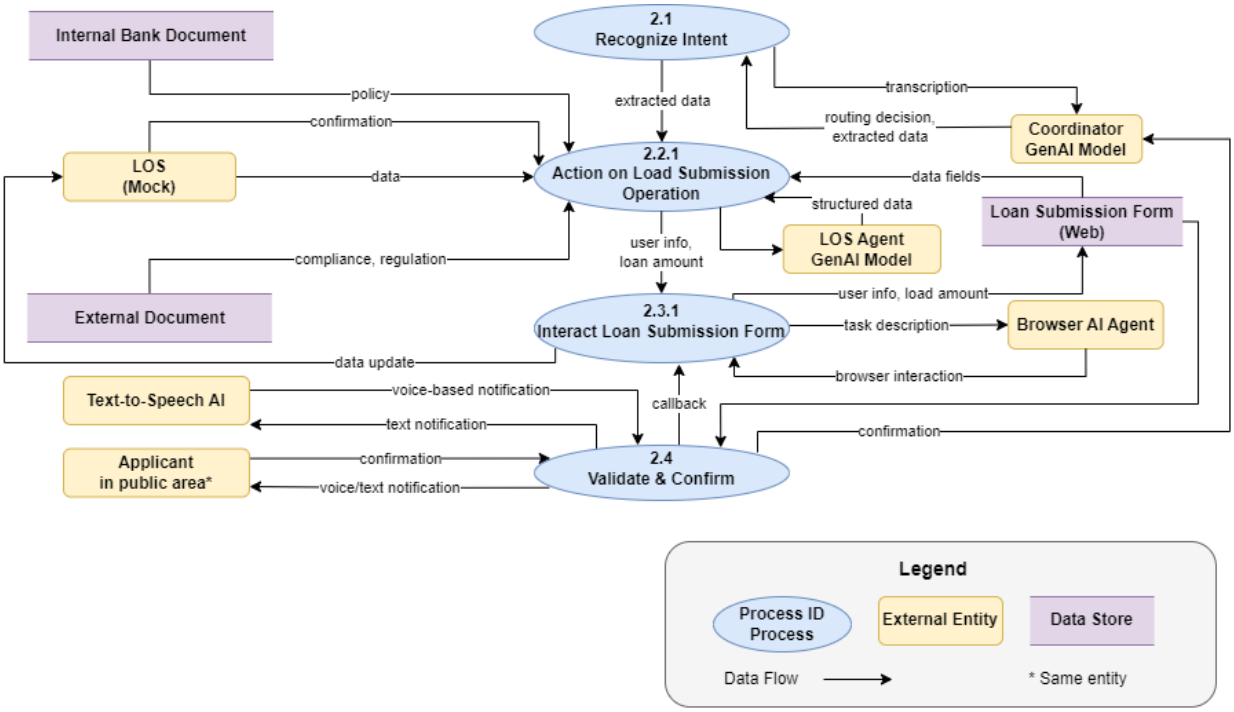


Figure 4: Level 3 Data Flow Diagram - Loan Submission

User Role: *Loan Applicant*

Workflow Description:

The loan submission workflow enables customers to apply for loans using natural voice commands. The system recognizes intent to apply for a new loan or update existing loan details, extracts key information (loan type, amount, term, applicant details), validates against internal policies and regulatory requirements, and automatically completes the online loan application form at vayonline.vpbank.com.vn.

Process Flow:

1. **Intent Recognition:** Silero VAD detects speech, Smart-Turn manages dialogue context, Coordinator GenAI identifies loan application intent

2. **Loan Operations:** LOS Agent structures loan data, cross-references internal policies and external compliance docs, validates request against regulations

3. **Form Interaction:** Browser-Use Agent navigates to loan form, inputs applicant details, selects dropdown options, submits application

4. **Validation & Confirmation:** System verifies submission success, updates LOS database and compliance records, provides voice confirmation via TTS

6.4.2 CRM Customer Update Workflow

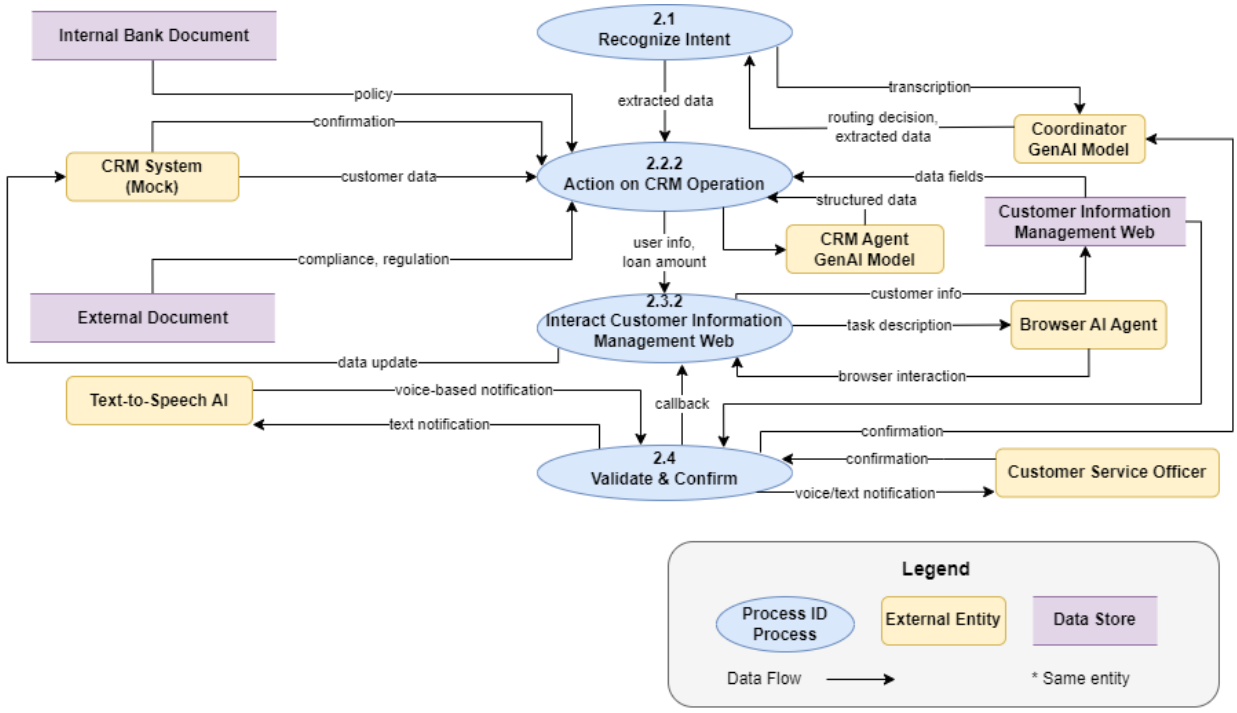


Figure 5: Level 3 Data Flow Diagram - CRM Customer Update

User Role: *Customer Service Officer*

Voice-driven updates to customer information including address changes, contact detail modifications, and relationship status updates. The CRM Agent ensures data consistency across all VPBank customer touchpoints.

6.4.3 HR Job Posting Workflow

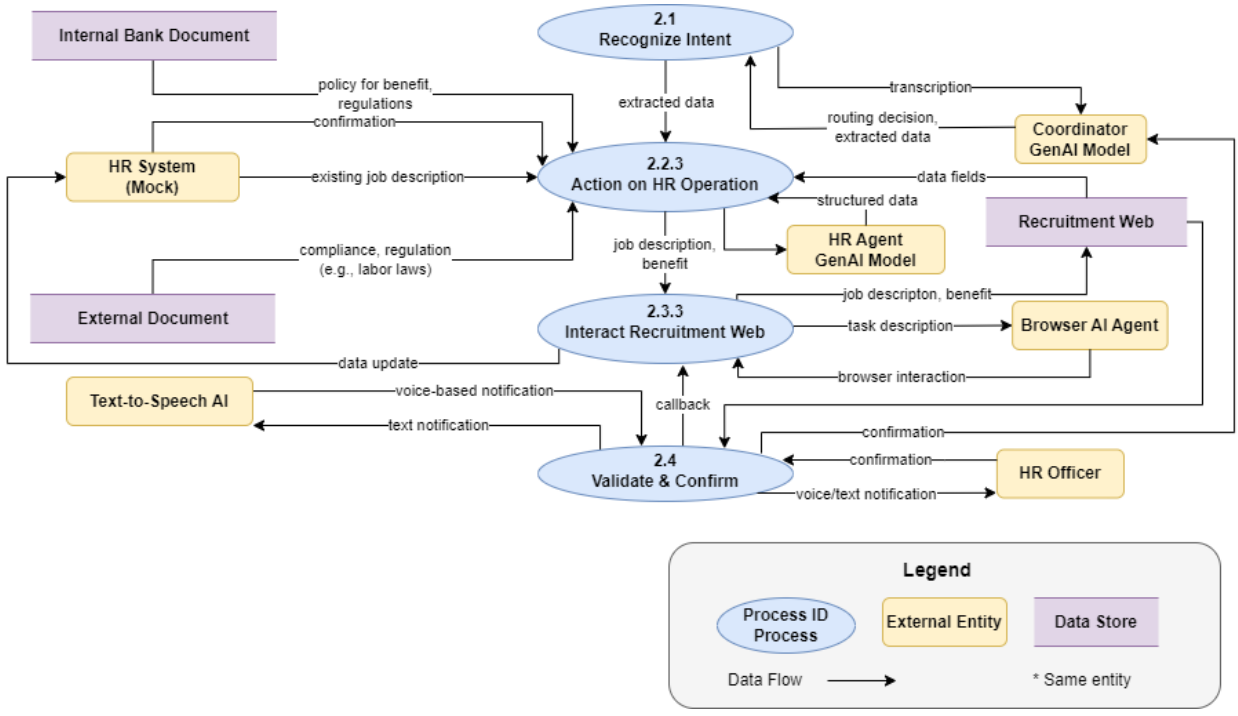


Figure 6: Level 3 Data Flow Diagram - HR Job Posting

User Role: *HR Officer*

Automated job description creation and posting to recruitment platforms. HR Agent extracts job requirements, validates against internal policies, and publishes to career portals.

6.4.4 Compliance Validation Workflow

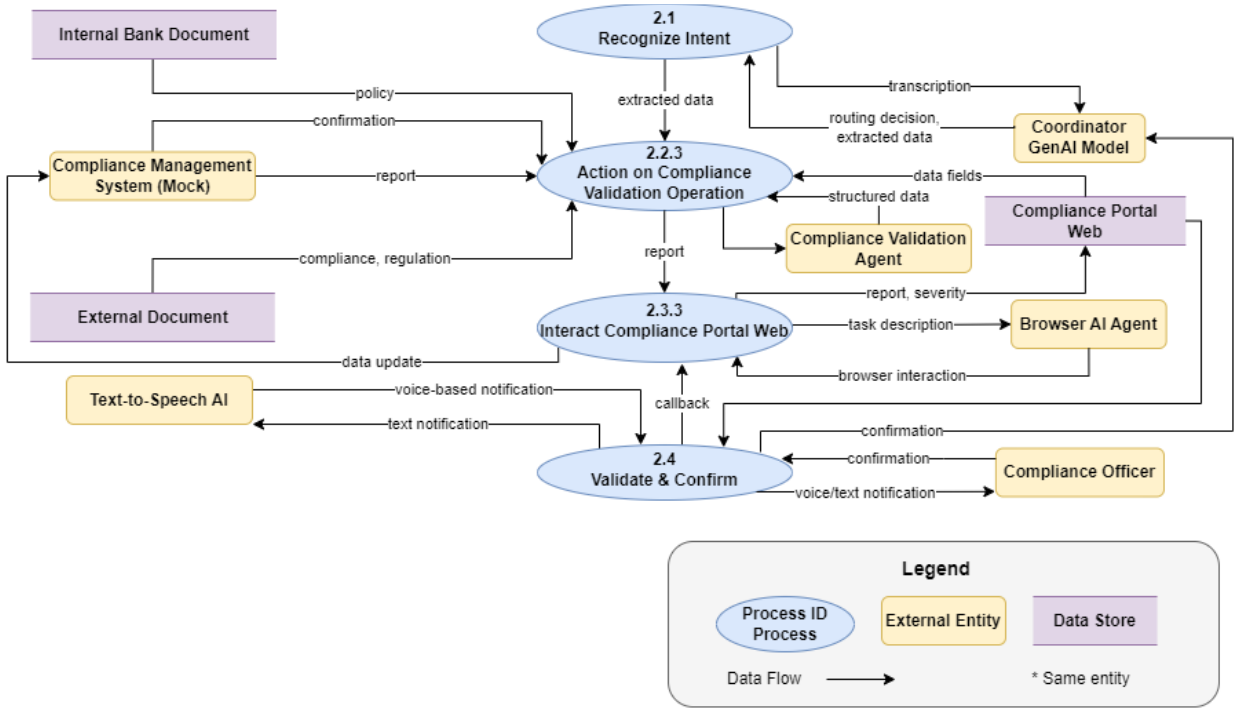


Figure 7: Level 3 Data Flow Diagram - Compliance Validation

User Role: *Compliance Officer*

Voice-initiated compliance report generation and submission. Compliance Agent cross-checks against regulatory frameworks (UCP 600, ISBP 821), validates data integrity, and submits to compliance portal with full audit trail.

7. Component Design

The VPBank Speak-to-Input system implements a modular component architecture where each component has clear responsibilities, well-defined interfaces, and manageable dependencies. This section details the design of key components including their technical implementation, integration points, and operational characteristics.

Component	Technology
Voice Pipeline	Pipecat, WebRTC, Silero VAD
Speech Recognition	PhoWhisper (fine-tuned), SageMaker
Workflow Router	Amazon Bedrock Claude, LangGraph
Domain Agents	LangChain, Amazon Bedrock
Browser Automation	Browser-Use
Text-to-Speech	ElevenLabs API

7.1 Voice Pipeline Component

7.1.1 Component Overview

Attribute	Value
Component Name	Voice Pipeline Engine
Primary Technology	Pipecat Framework, WebRTC, AI, Silero VAD
Programming Language	Python 3.11, JavaScript (WebRTC client)
Deployment	AWS ECS Fargate (containerized), Auto-scaling enabled
Key Dependencies	API Gateway, PhoWhisper STT, ElevenLabs TTS, DynamoDB State Store

7.1.2 Core Responsibilities

- Real-time audio capture and streaming via WebRTC protocol
- Audio quality enhancement through Silero VAD noise cancellation
- Voice Activity Detection (VAD) using Silero models
- Turn detection for multi-party conversations
- Audio buffering and stream management
- Bidirectional communication: upstream audio and downstream TTS
- Session management and connection state tracking

7.2 Component Details – Client Layer

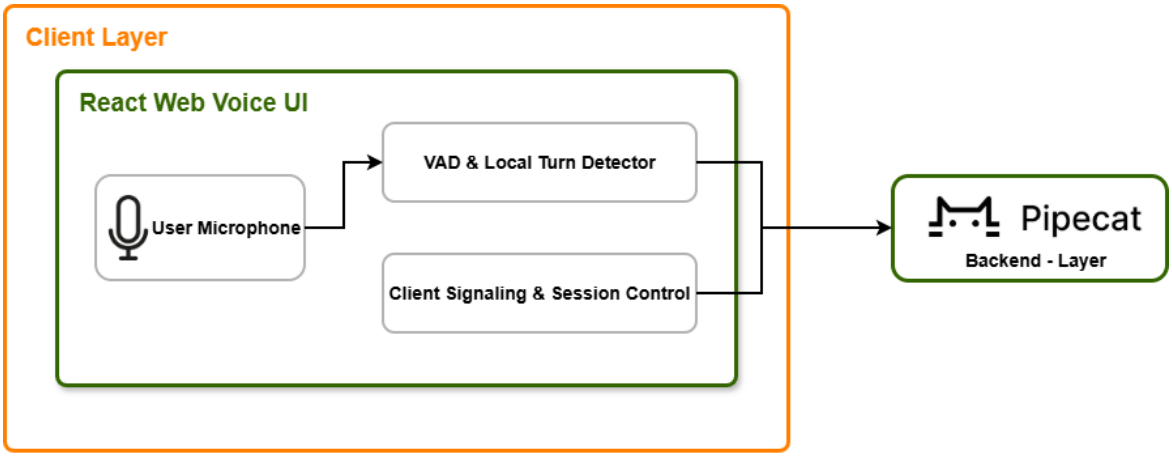


Figure 9: Client Layer Diagram

The client architecture orchestrates multiple layers – audio capture, local enhancement, activity detection, and real–time communication – before handing off to backend AI services. At the heart of this architecture lies a WebRTC–based audio pipeline, powered by **Pipecat's client SDK**. The browser initializes audio capture through the WebRTC getUserMedia() API, which opens a live audio stream from the user's microphone. This stream is encoded in a lightweight codec such as Opus or raw PCM and processed locally before being transmitted. WebRTC's built–in noise reduction and echo cancellation provide a first level of cleanup, but to achieve production–grade clarity, the system integrates **Pipecat's Krisp feature** for AI–powered noise suppression.

Krisp runs as an **on–device WebAssembly audio processor**, which means that the audio never leaves the browser for preprocessing. Instead, Krisp intercepts the raw microphone stream, identifies and removes environmental noise – such as keyboard clicks, fan hum, or background chatter – while preserving the natural tone and intelligibility of the speaker's voice. This real–time enhancement step dramatically improves both the transcription accuracy of downstream Speech–to–Text models and the overall responsiveness of the system. Because Krisp operates locally, it also strengthens privacy and reduces cloud processing costs.

Once the audio is cleaned, it flows into the **Voice Activity Detection (VAD)** module. The VAD continuously analyzes short segments of the audio stream to determine whether the user is actively speaking or silent. When speech is detected, the system begins streaming audio frames to the backend; when silence is sustained for a certain threshold, it automatically pauses transmission. This intelligent gating mechanism avoids sending unnecessary silent data to the backend, which optimizes both cost and bandwidth usage. It also enables a more natural user experience – the user simply speaks, pauses, and continues without needing to manually trigger recording.

When speech is active, audio frames are packaged and transmitted through a **secure WebRTC** connection to the Pipecat gateway. WebRTC are essential for real–time, bidirectional communication: the client sends binary audio chunks upstream while simultaneously receiving transcription updates, intent parsing, and AI–generated responses downstream. The connection is persistent and event–driven, ensuring millisecond–level latency between user speech and the corresponding AI feedback.

On the user interface, this architecture supports live transcription visualization – partial text updates appear on the screen as the model processes each segment. The client also renders playback of synthesized speech responses (from AWS Polly or other TTS providers) and can animate “listening” or “thinking” indicators based on the VAD state. The entire flow is orchestrated through React hooks or equivalent frontend logic, using the Pipecat client SDK to handle Krisp activation, VAD thresholds, and WebRTCsession management.

Security is embedded throughout the client pipeline. All data is transmitted over encrypted **WebRTC** channels, and each session includes authentication metadata (such as JWTs or signed tokens). Importantly, because Krisp operates entirely within the user’s device, no raw voice data is stored or transmitted externally during preprocessing.

7.3 Component Details – Voice Processing

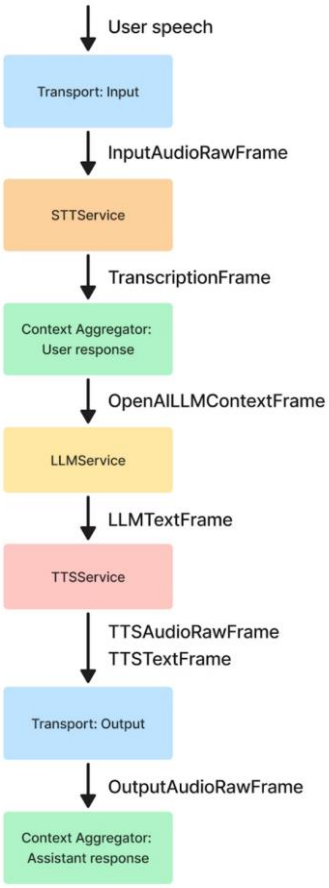


Figure 10: Detailed Voice Processing Layer Diagram

The **Voice Pipeline** serves as the entry point of the conversational system, enabling real–time, low–latency voice interaction between users and backend AI agents. Built on Pipecat, an open–source audio orchestration framework, this layer integrates multiple subcomponents to handle streaming, detection, transcription, and preprocessing efficiently.

Pipecat Framework: Pipecat acts as the core orchestrator for audio data flow. It manages:

- Audio streaming and buffering across the network
- Bidirectional WebRTC connections
- Real–time synchronization with AI inference layers

Through its modular pipeline design, Pipecat ensures that each processing unit (VAD, STT, TTS, and Agent Orchestration) can be independently optimized or replaced without affecting overall stability.

WebRTC Connection: The client (web or mobile) communicates with the server using WebRTC, which Pipecat handles natively. This ensures:

- Low–latency, full–duplex voice transmission

- End–to–end encryption (SRTP/TLS)
- Adaptive bitrate streaming, optimizing audio quality under fluctuating network conditions

PhoWhisper Speech–to–Text (STT): After VAD filtering, the audio stream is routed to fine-tuned PhoWhisper, a Vietnamese–optimized speech–to–text model derived from **OpenAI Whisper.** The fine-tuned progress is described in section 1.5 – Fine-Tune Automatic Speech Recognition Model Detail in "ARCHITECTURE OF SOLUTION".

The model provides**:**

- High accuracy for tonal languages, preserving diacritics and phonetic nuances in Vietnamese.
- Dialect robustness, handling **Northern, Central, and Southern** variations fluently.
- Banking domain adaptation, recognizing terminology specific to loan origination, CRM updates, and HR workflows.

Deployment Flexibility and Data Governance

Unlike most commercial transcription APIs, PhoWhisper supports on–premise or private–cloud deployment, ensuring:

- Full data sovereignty – no customer audio leaves the organization's controlled environment.
- Compliance with internal and regional banking regulations.
- Customizable fine–tuning on proprietary datasets using AWS SageMaker for continuous improvement.

Domain and Accent Adaptation, PhoWhisper is continually improved through:

- Domain–specific fine–tuning, using banking, compliance, or HR datasets for contextual understanding.
- Accent adaptation, trained on Vietnamese regional speech samples to improve recognition accuracy across user demographics.
- Feedback loop integration, where anonymized user commands are periodically aggregated and used for monthly model refinement.

These adaptation mechanisms are critical to sustaining long–term accuracy and user trust in voice–driven enterprise automation.

Open–Source Advantage

PhoWhisper and Pipecat, being open–source frameworks, provide flexibility beyond commercial lock–ins.

- Extend pipeline logic (e.g., integrate Krisp for noise suppression).
- Adjust VAD sensitivity or transcription thresholds.
- Customize downstream routing to proprietary workflow agents.

7.4 Component Details – AI Agent System

The Voice–AI automation framework is designed as a modular multi–agent system, where each agent specializes in a distinct enterprise function. The **Workflow Router Agent** orchestrates the overall process, routing requests to task–specific agents – including the Loan, CRM, HR, Compliance, and Browser Execution agents – ensuring seamless task automation, security, and compliance.

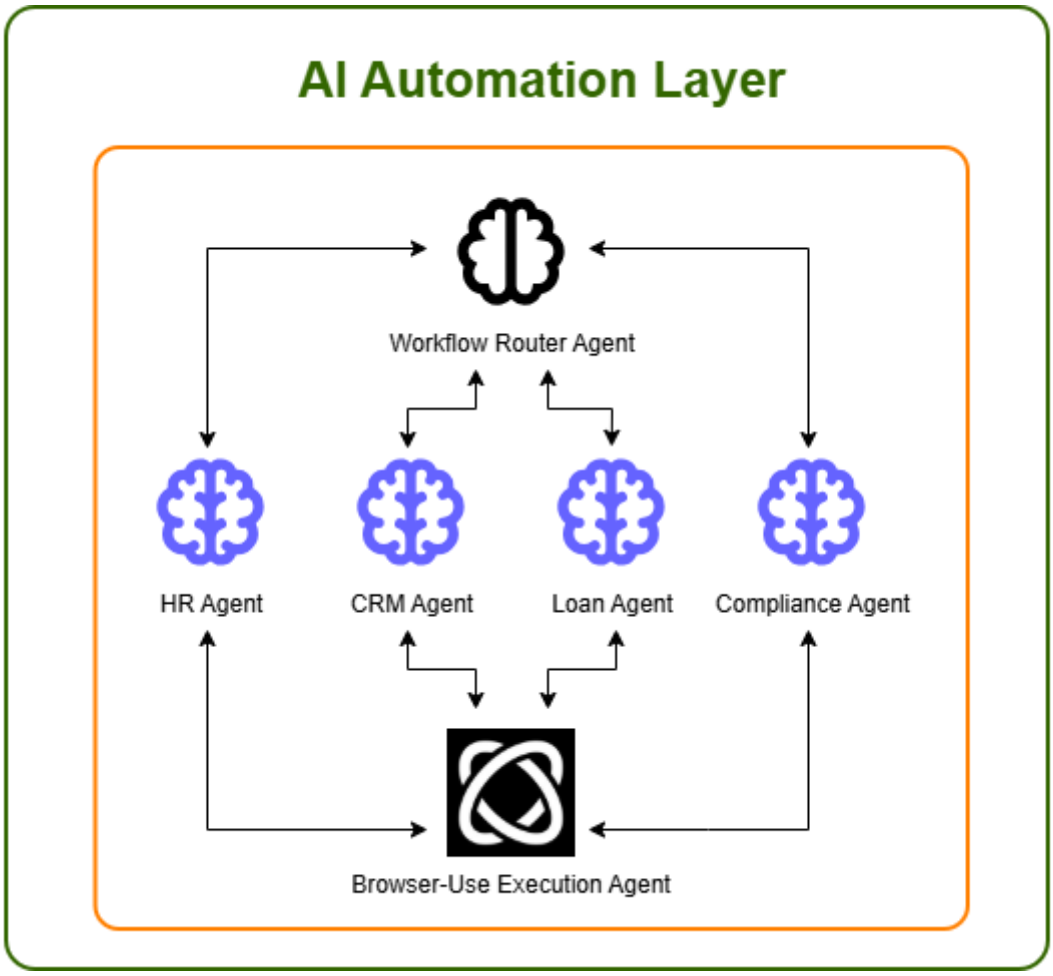


Figure 11: Multi–Agent Layer Diagram

Workflow Router Agent

This is the **core orchestration agent**, responsible for **intent recognition**, **context management**, and **inter–agent coordination**. It receives natural language inputs (voice or text), identifies the corresponding business domain, and delegates the task to the appropriate specialized agent.

Example workflow: A user says: “Fill in customer Nguyễn Văn An, loan amount 500 million, term 24 months.” → The Router Agent detects a *Retail Lending* intent. → It then dispatches the request to the **Loan Agent** for structured processing.

Key functions:

- Intent classification and routing
- Context and session management
- Agent–to–agent communication via API, HTTP, or WebRTC
- Logging, audit trail generation, and compliance reporting hooks

Compliance Agent

Handles **Use Case 4 – Risk, Compliance & Audit Automation**. This agent ensures all automated workflows comply with regulatory and institutional policies, particularly in AML/KYC reporting and data governance.

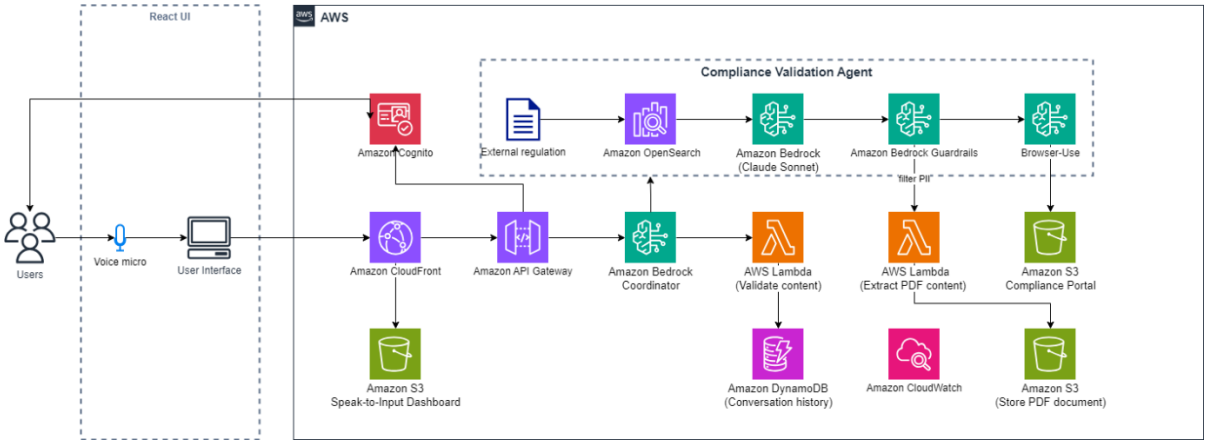


Figure 12 – Compliance Agent Architecture

(LOS agent, CRM agent and HR agent has the same architecture)

Functions:

- Automates report generation for AML, KYC, and regulatory submissions
- Cross–validates extracted data against internal systems for consistency
- Logs every action with a timestamp and user signature for auditability
- Enforces compliance with GDPR, PDPA, and SBV (State Bank of Vietnam) data retention regulations

Example:

A compliance officer states: “Generate September AML report, check completion status, and verify no violations.” → The Compliance Agent retrieves data, fills the report, validates entries, and archives all logs for audit trail compliance.

The Compliance Validation Agent implements a comprehensive AWS serverless architecture designed for automated document compliance filling and checking against regulatory standards and organizational policies.

Workflow Process

Frontend Layer

- React UI provides the user interface for reading the compliance report.
- Amazon CloudFront delivers the application with global distribution.
- Amazon S3 hosts static web assets.

API and Authentication Layer

- Amazon API Gateway manages API requests and routing to backend services.
- Amazon Cognito handles user authentication and role-based access control.
- Ensures secure document submission and compliance report access.

Document Processing Pipeline

- AWS Lambda (Extract PDF content) processes uploaded documents for compliance analysis.
- Amazon S3 (Store PDF document) securely stores original documents with audit trails.

AI-Powered Compliance Validation

- AWS Lambda (Validate content) orchestrates the compliance checking process.
- Knowledge Base contains regulatory frameworks and compliance rules:
- Amazon Bedrock Guardrails enforce content safety and compliance boundaries
- Amazon Bedrock (Claude 4.5 Sonnet) performs intelligent document analysis
- Browser-Use to interact with the compliance portal website.
- Amazon OpenSearch enables semantic search across compliance databases

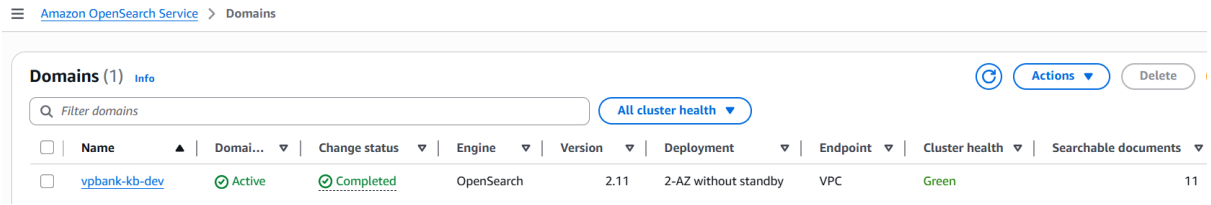


Figure <numb>: OpenSearch for indexing documents

Compliance Analysis Engine

The validation engine performs:

- Regulatory Compliance Checking: Validates documents against industry standards (UCP 600, ISBP 821,...).
- Policy Adherence Verification: Ensures alignment with organizational policies.
- Risk Assessment: Identifies potential compliance violations and risk levels.
- Gap Analysis: Highlights missing required elements or documentation.

Data Storage and Audit Layer

VPBank Voice Agent uses Amazon DynamoDB to store session history, conversation transcripts, and workflow execution logs. All data is stored with a TTL (Time To Live) of 90 days for automatic cleanup.

Monitoring and Observability

- Amazon CloudWatch provides comprehensive monitoring, alerting, and audit logging.
- Real-time compliance dashboard and automated notifications for violations.

Loan Agent

Handles **Use Case 1 – Loan Origination & KYC Automation**. It converts spoken instructions into structured loan application data and ensures pre–submission validation.

Functions:

- Parses natural language commands from Relationship Managers (RMs)
- Extracts entities such as *customer name*, *loan amount*, and *term length*
- Interfaces with the **Browser–Use Execution Agent** to fill loan origination forms in the LOS system

Requests confirmation and correction before final submission

CRM Agent

Handles **Use Case 2 – CRM Update & Customer Interaction Logging**. It keeps customer data synchronized across systems without manual entry.

Functions:

- Interprets natural commands, e.g., “Update customer Nguyễn Văn Bình’s address to 25A Nguyễn Trãi.”
- Executes updates via CRM APIs or browser automation
- Logs all interactions and updates automatically to ensure traceability

HR Agent

Handles **Use Case 3 – HR & Internal Workflow Automation**. It automates employee self–service actions and HR form processing.

Functions:

- Parses input like “Create a leave request from Oct 22 to 24 for personal reasons.”
- Identifies the relevant HR form, fills and validates data
- Submits through **Browser–Use Execution Agent**, with confirmation sent to the employee
- Ensures role–based access control (RBAC) for internal compliance

Browser–Use Execution Agent

This is the **execution layer** of the automation framework – effectively the system’s “hands.” It performs browser–level actions securely through sandboxed environments

Functions:

- Automates form–filling, button clicks, and file uploads
- Interacts with enterprise portals (LOS, CRM, HRIS, Compliance dashboards)
- Captures screenshots and action logs for traceability
- Operates within a secure, isolated environment with TLS encryption

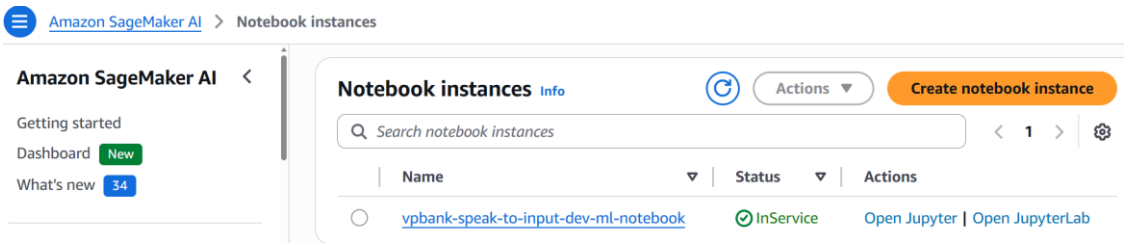
Context Management and Inter-Agent Communication

In this multi-agent system, The Workflow Router Agent serves as the central orchestrator, managing a shared context that stores all session information (history, user intent, collected data like customer name or loan amount). When a user makes a request, the Router Agent analyzes it, attaches the relevant context, and delegates the task to a **specialized agent** (e.g., Loan Agent, CRM Agent). That agent performs its function, updates the context with new data (e.g., “KYC status: verified”), and returns the result to the Router Agent. This ensures all agents operate from a **single, consistent source of truth**, enabling smooth handling of complex, multi-step workflows without losing information.

7.5 Speech Recognition Component (PhoWhisper)

7.5.1 Fine-Tuned Model Specifications

Attribute	Value
Model Base	Pho Whisper-Small (fine-tuned)
Model Name	PhoWhisper-small-v3-banking
Training Dataset	12 hours Vietnamese speech with regional accents
Deployment	AWS SageMaker (ml.g4dn.xlarge GPU instances)



7.5.2 Technical Specifications

[DIAGRAM: PhoWhisper Fine-Tuning Pipeline]

[Code fine-tune model]

8. API & Schema Definition

This section documents all REST API endpoints, message schemas, and data structures. All APIs follow OpenAPI 3.0 specification standards and implement RESTful design principles.

8.1 DynamoDB Schema Definitions

Table: `vpbank-sessions`

Primary Key:

- **Partition Key:** `session_id` (String) - Unique session identifier

Attributes:

Attribute	Type	Description	Required
`session_id`	String	Unique session identifier (format: `YYYYMMDD_HHMMSS`)	✅ Yes
`started_at`	String	ISO 8601 timestamp when session started	✅ Yes

ended_at	String	ISO 8601 timestamp when session ended	❌ Optional
created_at	Number	Unix timestamp (seconds) when record was created	✅ Yes
messages	List	Array of message objects (user/assistant exchanges)	✅ Yes
workflow_executions	List	Array of workflow execution records	❌ Optional
ttl	Number	Unix timestamp (seconds) - TTL 90 days from created_at	✅ Yes

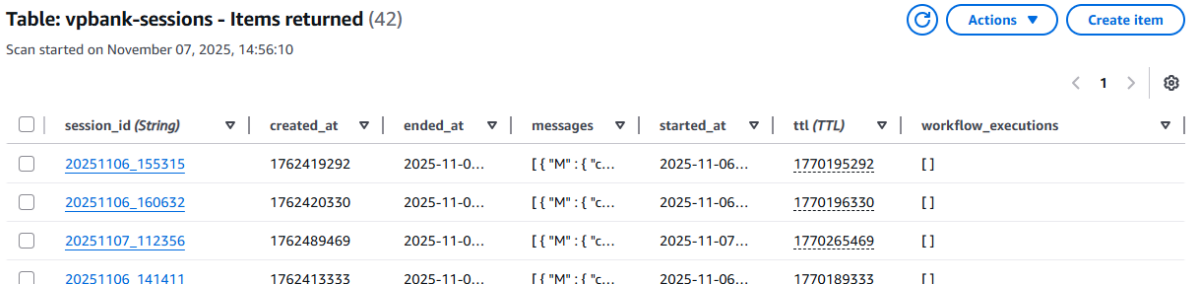


Table: vpbank-sessions - Items returned (42)

Scan started on November 07, 2025, 14:56:10

Actions ▼ | Create item

1

session_id *(String)*	created_at	ended_at	messages	started_at	ttl *(TTL)*	workflow_executions
20251106_155315	1762419292	2025-11-0...	[{"M":{"c...	2025-11-06...	1770195292	[]
20251106_160632	1762420330	2025-11-0...	[{"M":{"c...	2025-11-06...	1770196330	[]
20251107_112356	1762489469	2025-11-0...	[{"M":{"c...	2025-11-07...	1770265469	[]
20251106_141411	1762413333	2025-11-0...	[{"M":{"c...	2025-11-06...	1770189333	[]

8.2 API <1>

[API SCHEMA 1]

[attached rendered image for api schema 1]

9. Evaluation & Testing

Comprehensive testing framework to ensure the VPBank Speak-to-Input system meets all quality, performance, and business requirements before production deployment.

9.1 Evaluation Metrics

[TABLE: metrics, nhiều quá thì quăng link gg sheet]

9.2 Testing Strategy

9.2.1 Unit Testing

Scope: Individual component validation

- Voice pipeline: Audio processing accuracy, VAD precision/recall
- STT engine: WER on standard test sets, accent handling
- Intent recognition: Classification accuracy, entity extraction F1-score
- Browser automation: Form filling success rate, error detection

WebRTC Connection

[CODE WebRTC và output]

11/7/2025

Intent Detection from Keywords

[table vầy,

nhiều test case thì link gg sheet]

Test Case UT-VB-003: Intent Detection from Keywords
Component: VoiceBot.detect_form_intent()

Objective: Verify form intent is correctly detected from user utterance

User Message	Expected Intent	Status
Tôi muốn điền thông tin vay tiền	loan_form	PASS
Cập nhật thông tin khách hàng	crm_form	PASS
Xin chào	None	PASS

[1 (vài) Image minh chứng output]

9.2.2 Integration Testing

Scope: End-to-end workflow validation across components

- Voice → STT → Intent → Agent → Browser → Confirmation flow
- API integration: REST endpoints, WebSocket connections

- External system integration: mock banking websites, TTS services

9.2.3 User Acceptance Testing (UAT)

Participants: 11 AWS community members (4 people from the north, 3 people from the south, 4 people from the central region)

Duration: 1-week pilot period with structured test scenarios

1. Scenario 1 - Loan Application: Complete loan form via voice for <number> different customer profiles
2. Scenario 2 - CRM Update: Update customer address and contact info using voice commands
3. Scenario 3 - HR Job Posting: Create and publish job listing via voice
4. Scenario 4 - Compliance Report: Generate AML report using voice-driven workflow

Success criteria:

- **Task completion rate >85%**
- **User satisfaction score >4.0/5.0**
- **Zero critical bugs**
- **<5 high-priority bugs**

10. Cost & ROI Analysis

This section provides comprehensive cost breakdown, ROI calculations, and financial projections for the VPBank Speak-to-Input system. All estimates are based on industry benchmarks and AWS pricing as of November 2025.

10.1 AWS Service Cost Breakdown (for prototype)

Service Name	Description	Monthly Cost	Properties
Amazon CloudFront	Delivery static web content	$11.00	Data transfer out to internet: 100 GB per month Data transfer out to origin: 100 GB per month Number of requests (HTTPS): 500,000 per month

AWS Fargate	Self–host PhoWhisper	$27.26	Operating system: Linux CPU Architecture: x86 Average duration: 45 minutes Number of tasks or pods: 3 per day Amount of ephemeral storage: 50 GB
Workload 1	Core LLM	$86.40	Average requests per minute: 2 Hours per day at this rate: 1 Average input tokens per request: 500 Average output tokens per request: 1,500
Application Load Balancer		$22.27	Number of Application Load Balancers: 1
Amazon API Gateway		$0.07	HTTP API requests units: millions Average size of each request: 34 KB Requests: 20,000 per month
S3 Standard		$0.23	S3 Standard storage: 10 GB per month
VPN Connection		$0.00	Working days per month: 22
NAT Gateway		$32.85	Number of NAT Gateways: 1
Amazon CloudWatch		$25.23	Standard Logs: Data Ingested: 50 GB
Amazon Elastic Container Registry		$20.00	Amount of data stored: 200 GB per month
TOTAL		**$225.31**	

10.2 Production Scale Cost Estimation

10.2.1 Scaling Assumptions

- Transaction volume: 10,000 monthly transactions
- Peak capacity planning: 3x average for peak hours
- High availability: Multi-AZ deployment across availability zones
- Production-grade security and compliance requirements
- 99.9% uptime SLA requirement
- Enterprise monitoring and logging
- Disaster recovery and backup capabilities

10.2.2 AWS Service Cost Breakdown (Production Scale)

Service	Prototype	Production	Scale	Notes
CloudFront	$11.00	$55.00	5x	500 GB transfer, 2.5M requests
Fargate	$27.26	$327.12	12x	Continuous operation, 6 tasks
Bedrock Claude	$86.40	$1,036.80	12x	24 req/min, 8 hrs/day
Load Balancer	$22.27	$66.81	3x	Multi-AZ deployment
API Gateway	$0.07	$0.84	12x	240K requests/month
S3 Storage	$0.23	$11.50	50x	500 GB logs & audio
VPN Connection	$0.00	$73.00	New	2 VPN connections
NAT Gateway	$32.85	$98.55	3x	Multi-AZ deployment
CloudWatch	$25.23	$252.30	10x	500 GB logs
ECR	$20.00	$100.00	5x	1 TB storage
Database	$0.00	$285.00	New	PostgreSQL Multi-AZ DynamoDB
WAF Security	$0.00	$50.00	New	Web application firewall
ElastiCache	$0.00	$100.00	New	Redis caching
AWS Backup	$0.00	$30.00	New	Automated backups
KMS Encryption	$0.00	$15.00	New	Key management
Route 53	$0.00	$50.00	New	DNS management
AWS Config	$0.00	$40.00	New	Compliance monitoring
CloudTrail	$0.00	$30.00	New	Audit logging
Data Transfer	$0.00	$76.39	New	Cross-AZ & egress
TOTAL	**$225.31**	**$2,698.31**	**12x**	

10.3 Implementation Costs

Cost Category	Amount	Notes
Development & Integration	$120,000	Custom development, API integration
Testing & QA	$25,000	UAT, security, load testing
Training & Documentation	$15,000	Staff training, user guides
Infrastructure Setup	$13,000	AWS setup, networking, security
Subtotal Implementation	$173,000	One-time costs

Annual Operational Cost	$32,379.72	Production AWS costs

10.4 Operational Cost Savings

Baseline Manual Processing Costs

Cost Component	Monthly Cost	Calculation
Labor Cost	$10,000 - $20,000	500-1,000 staff hours × $20/hour
Error Correction	$3,000 - $5,000	1-4% error rate requiring rework
Transaction Processing	$50,000 - $100,000	10,000 transactions × $5-10 each
Total Baseline	$63,000 - $125,000	Conservative: $63,000/month

With AI Automation (Production Scale)

Cost Component	Monthly Cost	Calculation
Labor Cost (60% reduction)	$4,000 - $8,000	200-400 staff hours × $20/hour
Error Correction (<0.5%)	$200 - $400	Minimal rework needed
AI Transaction Cost	$100 - $200	10,000 × $0.01-0.02 per transaction
AWS Operational Cost	$2,698.31	Production infrastructure
Total Automated Cost	$6,998 - $11,298	Conservative: $11,298/month

10.5 ROI Calculation

Monthly & Annual Savings

Monthly Savings Calculation:

Baseline Cost: $63,000

Automated Cost: -$11,298

Monthly Savings: $51,702

Annual Savings: $620,424

First Year Financial Summary

Metric	Amount
Total Implementation Cost	$173,000
First Year Operational Cost	$32,380
Total First Year Cost	$205,380
Annual Cost Savings	$620,424
Net First Year Benefit	$415,044

ROI Metrics

First Year ROI:

ROI = (Net Benefit / Total Cost) × 100%

ROI = ($415,044 / $205,380) × 100%

ROI = 202%

Payback Period:

Payback = Total Cost / Monthly Savings

Payback = $205,380 / $51,702

Payback = 3.97 months (≈ 4 months)

10.7 Additional Business Value (Intangible Benefits)

- **Improved Customer Experience:** Form completion rate increases from 42-58% to 75-85%, enhancing customer satisfaction and retention
- **Faster Time-to-Market:** 73% reduction in processing time enables faster loan approvals and service delivery
- **Enhanced Accessibility:** 11.6% of population aged 60+ and 7.06% with disabilities gain improved access to banking services
- **Competitive Advantage:** First-mover advantage in Vietnamese voice-driven banking automation
- **Employee Satisfaction:** Reduced manual data entry workload improves job satisfaction and reduces turnover
- **Scalability:** System can scale to handle 5-10x current transaction volume without proportional cost increase

11. Security & Compliance

The VPBank Speak-to-Input system implements enterprise-grade security controls and compliance frameworks to protect sensitive customer data and meet Vietnamese banking regulatory requirements.

11.1 Security Architecture

[DIAGRAM: Security Architecture Overview]

Network layers, encryption boundaries, IAM roles, and audit trail flow

11.1.1 Authentication & Authorization

- **Amazon Cognito Integration:** Integrated with VPBank SSO (simulated) for single sign-on authentication

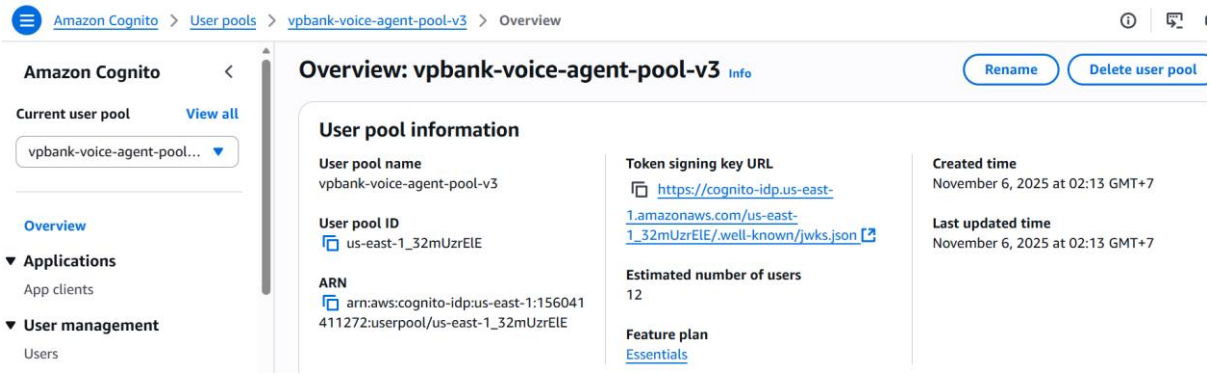


Figure <numb>: Amazon Cognitio user pool

- **Role-Based Access Control (RBAC):** Fine-grained permissions for different user roles (loan officers, CSRs, HR staff, compliance officers)
- **API Key Rotation:** Automatic rotation of API keys every day with AWS IAM role STS temporary credential rotation

11.1.2 Data Protection

- **Encryption at Rest:** AES-256 encryption for all data stored in S3, DynamoDB, and EBS volumes using AWS KMS customer-managed keys

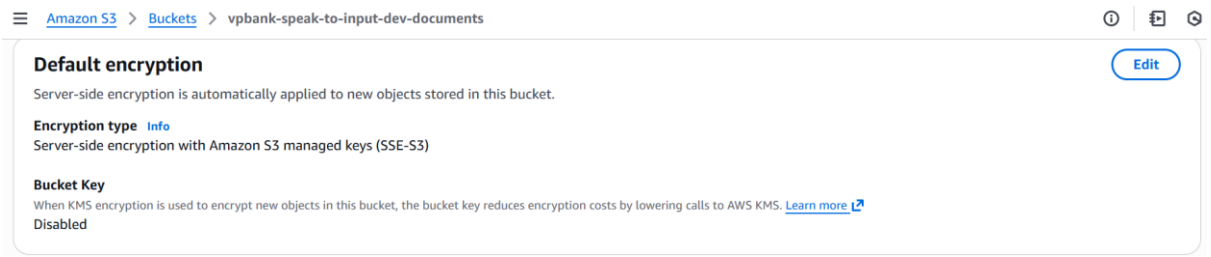
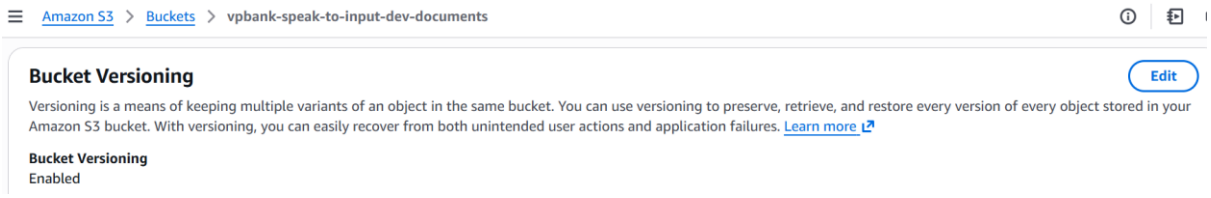
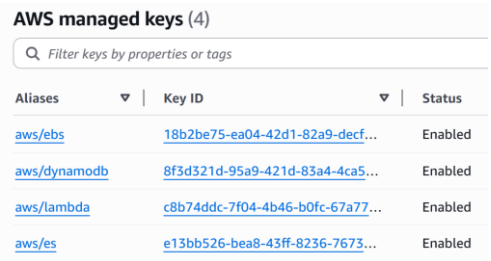


Figure <numb>: S3 AES-256 encryption with SSE-S3

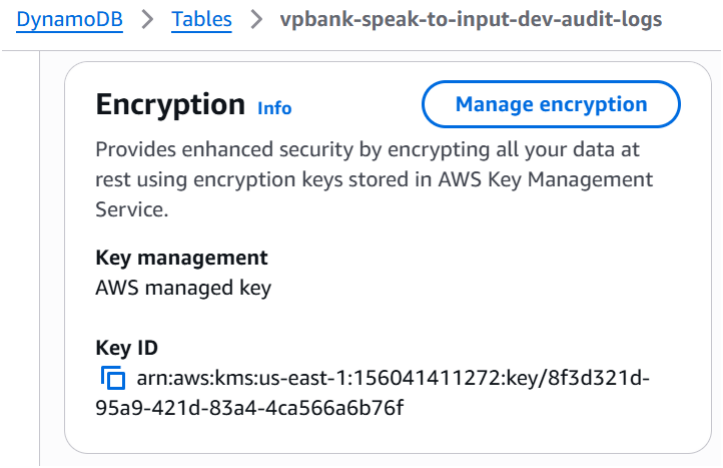
- **Enable S3 bucket versioning:** preserve, retrieve, and restore versions of objects, mitigates accidental deletions or overwrites



- **Key Management:** AWS KMS with automatic key rotation enabled, separate keys for different data classifications

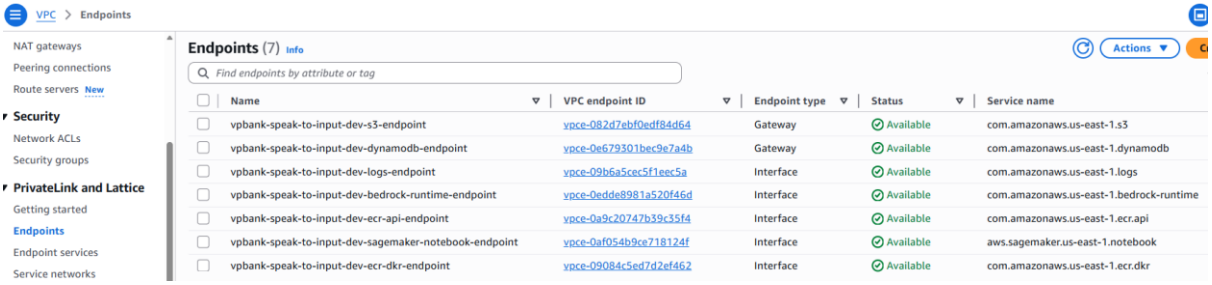


- **Database Encryption:** DynamoDB encryption at rest with AWS-managed keys

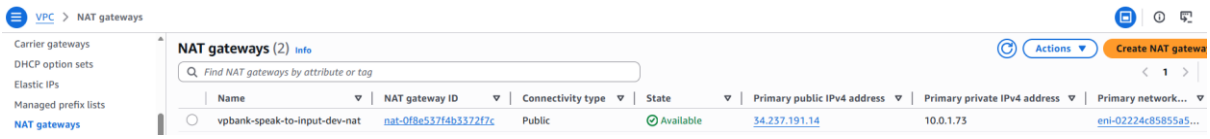


11.2 Network Security

- **VPC Isolation:** All services deployed in private subnets with no direct internet access
- **Security Groups:** Least-privilege firewall rules restricting traffic to necessary ports and protocols
- **VPC Endpoints:** Private connectivity to AWS services (Bedrock, S3, DynamoDB) without internet egress



- **NAT Gateway:** Controlled outbound internet access for external API calls (ElevenLabs, model updates)



11.3 Audit

Comprehensive audit logging ensures complete traceability of all system actions:

- CloudTrail LoggingAll AWS API calls logged
- Application LogsVoice interactions, transcriptions, intent recognition, and actions logged

User Activity TrackingEvery user action tied to authenticated identity with timestamp

12. Deployment Guide

This section provides step-by-step instructions for deploying the VPBank Speak-to-Input system using Infrastructure as Code (IaC) with Terraform and CI/CD automation via GitHub Actions.

12.1 Prerequisites

- AWS Account with administrator access
- Terraform Cloud account (for state management)
- GitHub account with Actions enabled
- AWS CLI configured with appropriate credentials
- Docker installed for local container builds
- kubectl installed for ECS task management
- Python 3.11+ and Node.js 18+ for local development

12.2 Deployment Architecture

AWS Architecture Implementation for the Voice-Driven Automation Platform implements the AWS architecture that powers our voice-driven automation platform. The design is engineered for scalability, security, and high availability, providing a robust foundation for the core features.

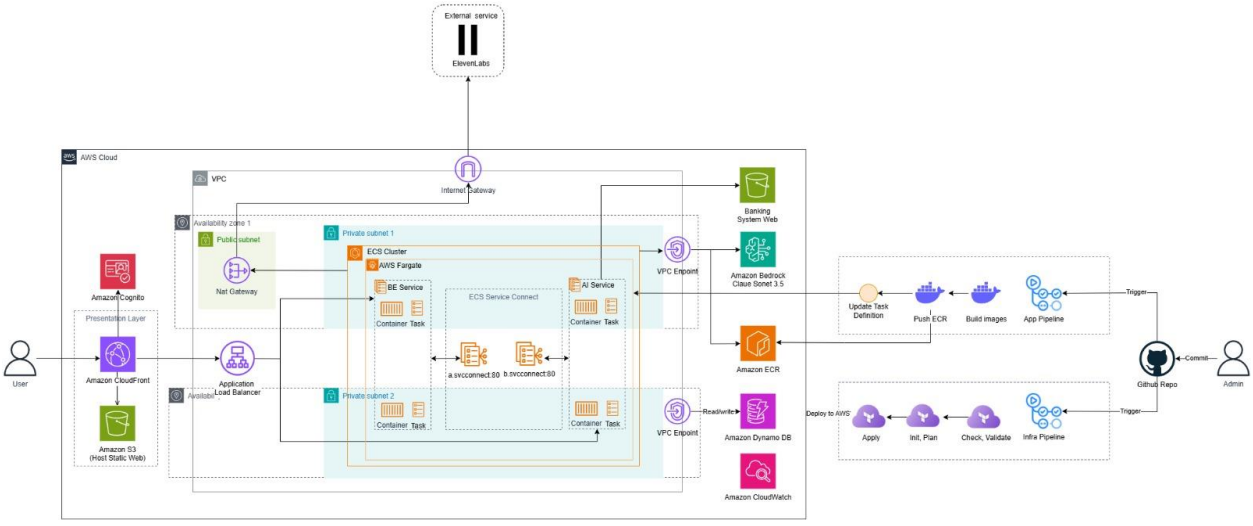


Figure 17: AWS Deployment Diagram

Infrastructure as Code Deployment: The cloud infrastructure is managed with Terraform using the Infrastructure as Code (IaC) approach for consistent, automated deployment. Terraform Cloud handles execution, state storage, locking, and centralized policy and access control across all environments.

Interface Delivery and Session Initiation: The user experience starts with a responsive web interface delivered via Amazon CloudFront and Amazon S3 for low-latency access. The interface captures voice input, manages the local session, and establishes a secure backend connection for real-time interaction.

User Authentication: User identity and access management are handled through Amazon Cognito, which provides a fully managed authentication and authorization solution for the application.

12.3 Infrastructure Deployment

12.3.1 Terraform Configuration

1. Clone Repository

2. Configure Terraform

```
# General Configuration
aws_region   = "us-east-1"
environment  = "dev" # dev | prod
project_name = "vpbank-speak-to-input"

# Networking
vpc_cidr             = "10.0.0.0/16"
availability_zones   = ["us-east-1a", "us-east-1b"]
public_subnet_cidrs  = ["10.0.1.0/24", "10.0.2.0/24"]
private_subnet_cidrs = ["10.0.11.0/24", "10.0.12.0/24"]

# DynamoDB
dynamodb_billing_mode         = "PAY_PER_REQUEST"
enable_point_in_time_recovery = true

# S3 Buckets
s3_bucket_names = {
  voice_recordings = "voice-recordings"
  documents        = "documents"
  frontend         = "frontend"
  artifacts        = "artifacts"
}

enable_s3_versioning = true

# ECS Configuration
ecs_task_cpu           = 1024
ecs_task_memory        = 2048
ecs_desired_count      = 1
enable_ecs_autoscaling = true

# SageMaker Configuration (for PhoWhisper)
sagemaker_instance_type  = "ml.g4dn.xlarge"
sagemaker_model_data_url = ""

# Lambda Configuration
lambda_runtime     = "python3.11"
lambda_memory_size = 512
lambda_timeout     = 60

# OpenSearch Configuration
opensearch_instance_type   = "t3.small.search"
```

```
opensearch_instance_count  = 1
opensearch_ebs_volume_size = 20

# CloudWatch
cloudwatch_log_retention_days = 90

# Feature Flags
enable_opensearch    = true   # Deploy OpenSearch for knowledge base
enable_sagemaker     = true   # Deploy SageMaker for PhoWhisper
enable_nat_gateway   = true   # Required for private subnet internet access
enable_vpc_endpoints = true   # VPC endpoints for cost optimization

# Additional Tags
additional_tags = {
  CostCenter = "VPBank-Hackathon-2025"
  Owner      = "Team-19"
  Hackathon  = "VPBank-Tech-Hack-2025"
}
```

3. Set Environment Variables

```
export AWS_ACCESS_KEY_ID="your_key"
export AWS_SECRET_ACCESS_KEY="your_secret"
export AWS_DEFAULT_REGION="us-east-1"
```

4. Initialize Terraform

terraform init

5. Review Infrastructure Plan

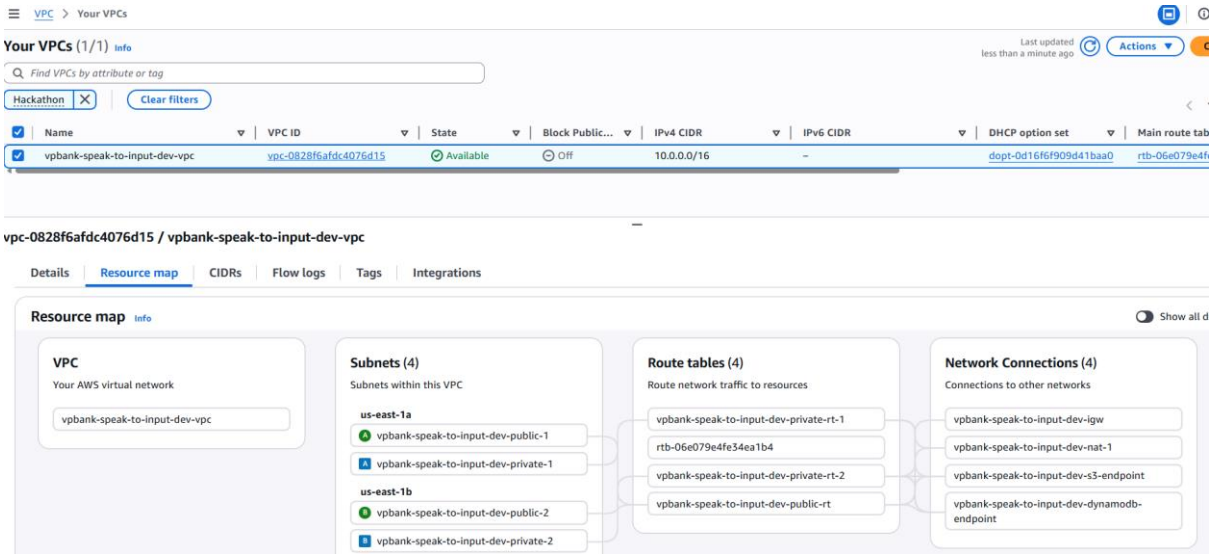
terraform plan -out=tfplan

6. Apply Infrastructure

terraform apply tfplan

12.3.2 Key Infrastructure Components Deployed

- VPC with public and private subnets across 2 Availability Zones



- ECS Fargate cluster for containerized services

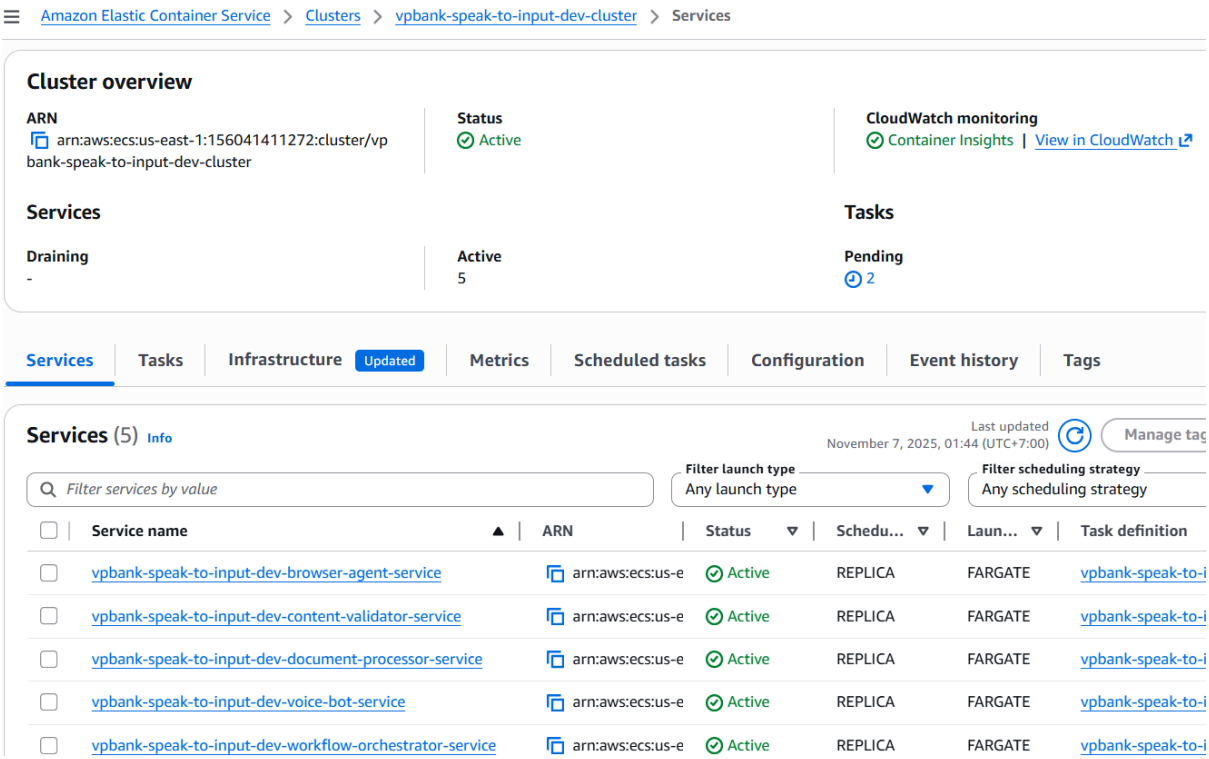


Figure <number>: ECS cluster

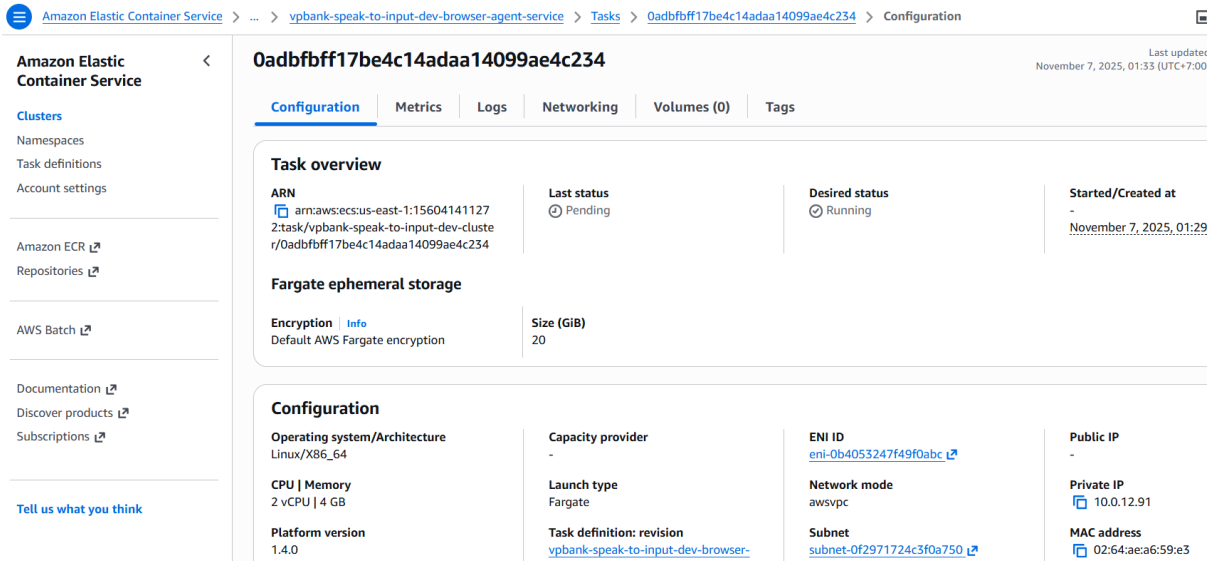
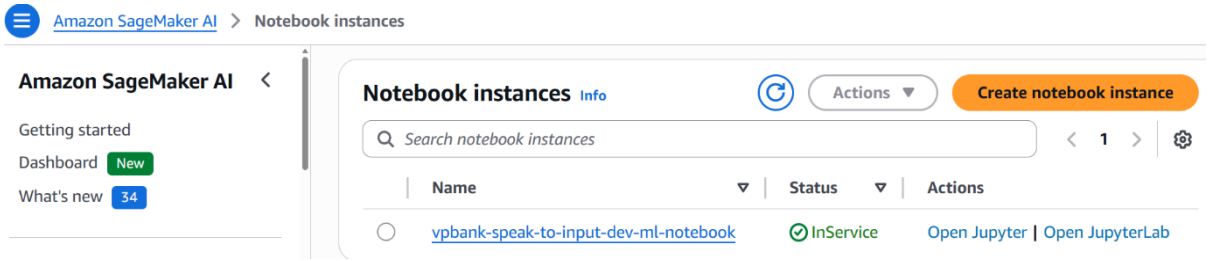


Figure <num>: ECS task infrastructure configuration

Figure <num>: ECS task definition

- SageMaker Jupyter Notebook for PhoWhisper model



- DynamoDB tables for session state and audit logs

[DynamoDB tables]

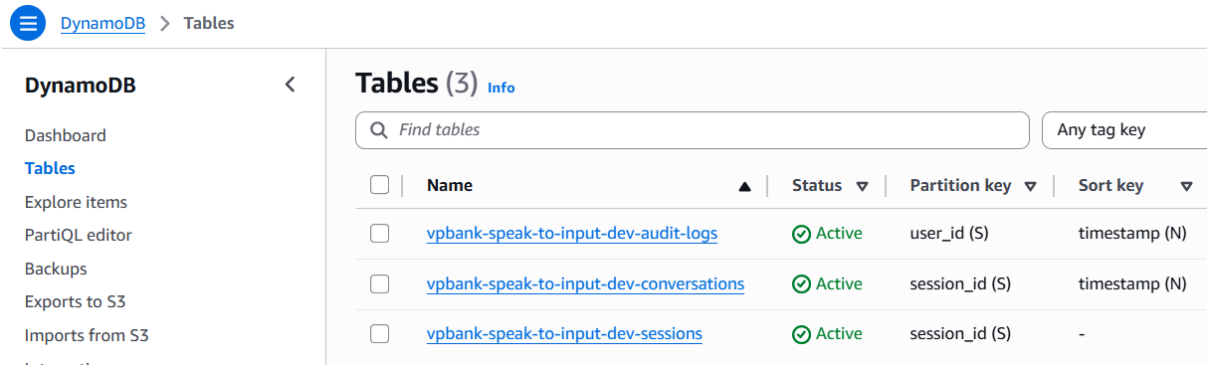


Figure <number>: audit log DynamoDB table

- S3 buckets for voice recordings and artifacts:

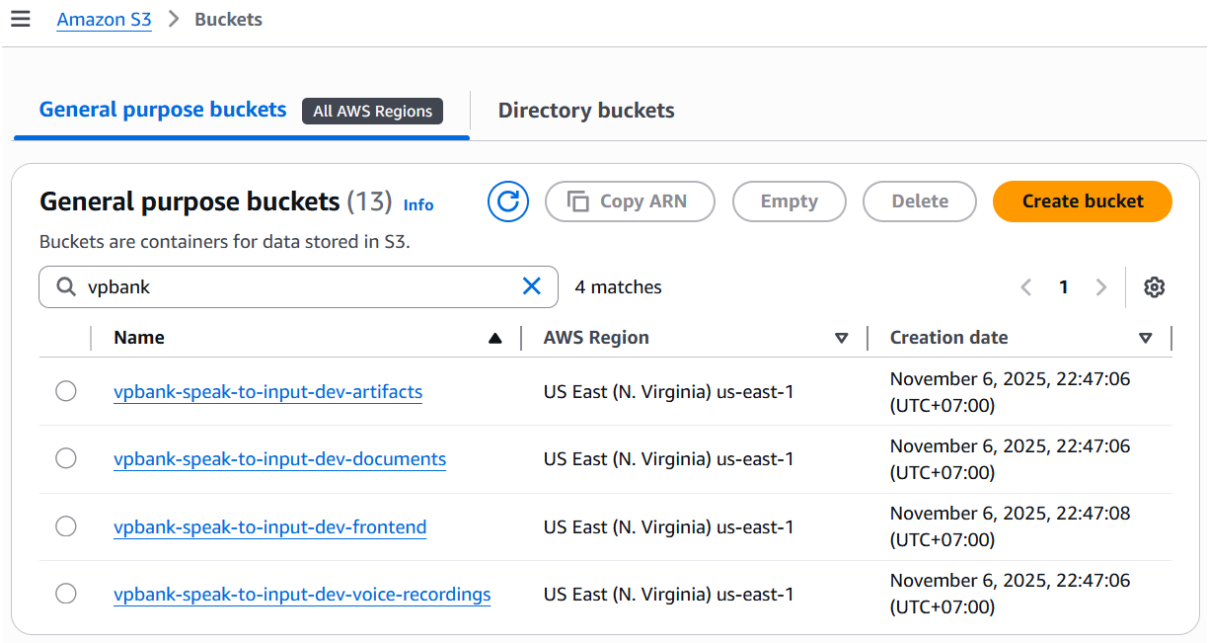


Figure <numb>: S3 buckets for voice recordings, artifacts and documents

- Amazon OpenSearch for indexing documents

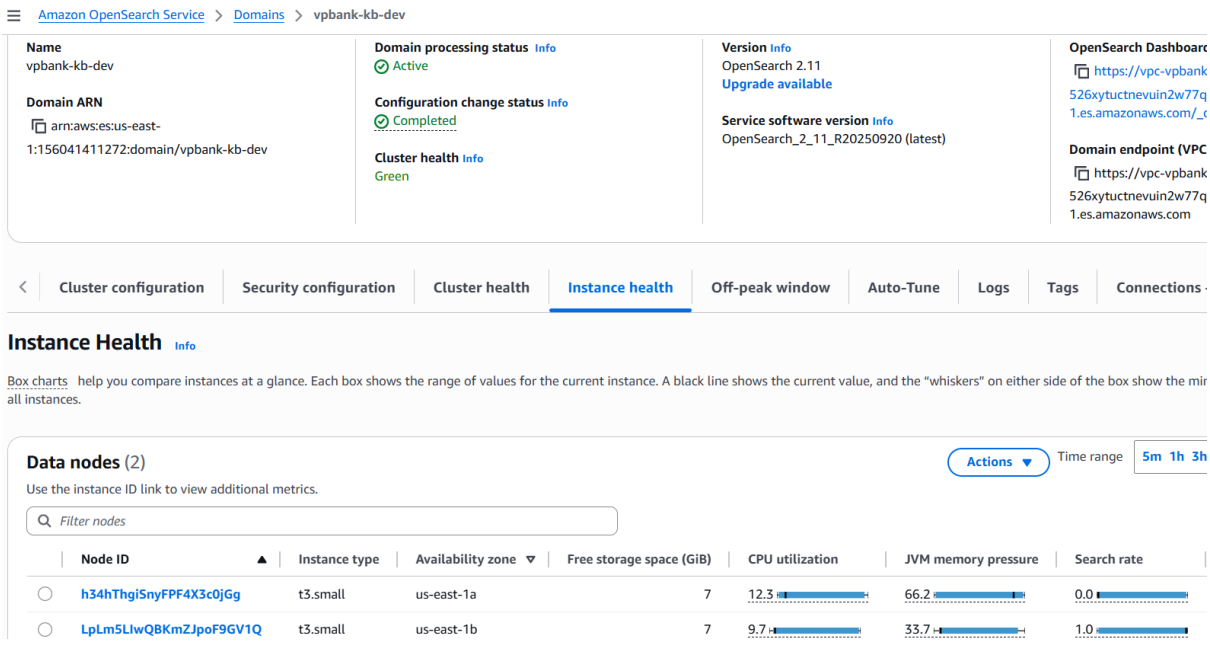


Figure <numb>: Infrastructure for Amazon OpenSearch domain

- CloudWatch log groups and dashboards

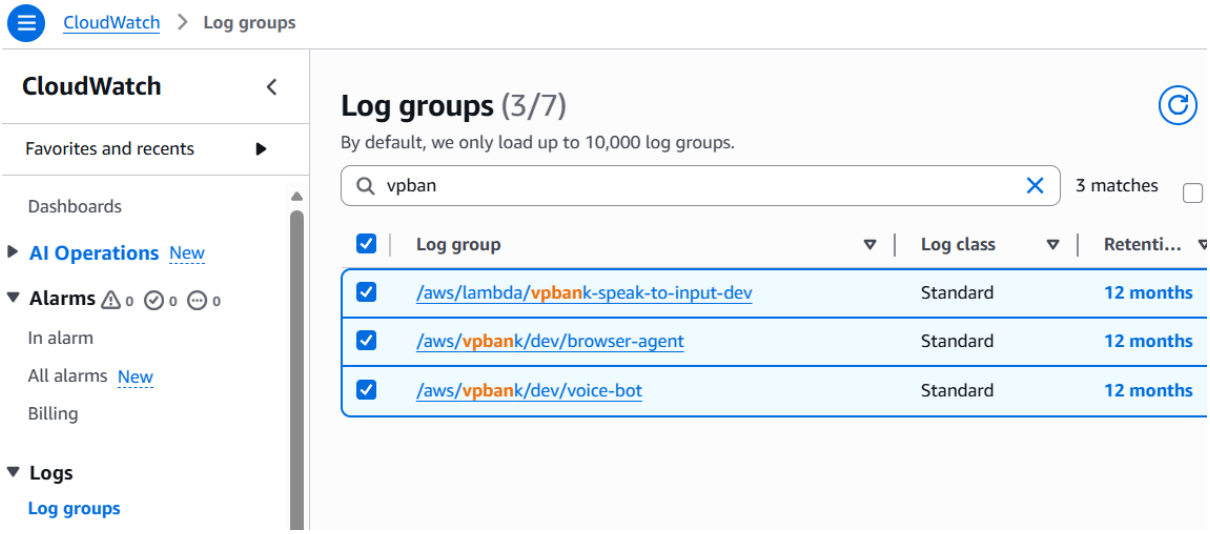


Figure <numb>: CloudWatch log groups

- API Gateway with custom domain and SSL certificate

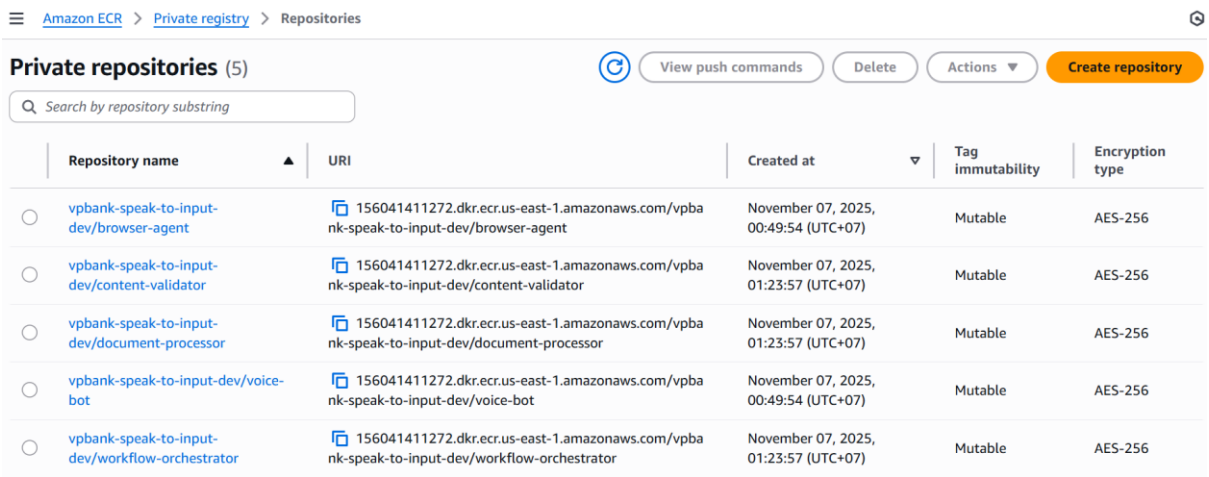
Figure <numb>: API Gateway

- Cognito user pool integrated with VPBank SSO for employees and customers (simulated):

12.4 Application Deployment

12.4.1 Container Image Build & Push

1. Build Docker images & tag
2. Push images to ECR private repositories



12.4.2 ECS Task Definition Update

ECS task definitions are automatically updated by Terraform when new images are pushed. Manual update can be performed via:

aws ecs update-service --cluster vpbank-speak-to-input-dev-cluster --service vpbank-speak-to-input-***-service --force-new-deployment

12.5 CI/CD Pipeline

GitHub Actions automates the build, test, and deployment process:

Pipeline Stages

- **Stage 1: Code Quality:** Linting, security scans
- **Stage 2: Build:** Docker image build and push to ECR
- **Stage 3: Deploy to Staging:** Automatic deployment to staging environment
- **Stage 4: Integration Tests:** Automated E2E tests in staging
- **Stage 5: Manual Approval:** Required approval for production deployment
- **Stage 6: Deploy to Production:** Blue-green deployment to production ECS cluster

12.6 Configuration Management

- **Environment Variables:** Stored in AWS Systems Manager Parameter Store
- **Secrets Management:** API keys and credentials in AWS Secrets Manager
- **Configuration Files:** Versioned in Git with environment-specific overrides

12.7 Rollback Procedures

1. Identify issue via CloudWatch alarms or user reports
2. Execute rollback command:

```
aws ecs update-service --cluster vpbank-speak-to-input-dev-cluster --service vpbank-speak-to-input-***-service --task-definition <PREVIOUS_VERSION>
```

3. Monitor rollback progress in ECS console
4. Verify system health
5. Notify stakeholders of rollback completion

13. Monitoring & Maintenance

Comprehensive monitoring and maintenance procedures ensure the VPBank Speak-to-Input system operates reliably, performs optimally, and continuously improves over time.

[DIAGRAM: Monitoring Architecture]

CloudWatch → SNS → Email team

13.1 System Monitoring

13.1.1 CloudWatch Dashboards

Infrastructure Health Dashboard: ECS task CPU/memory utilization, DynamoDB throttling, S3 request rates, API Gateway errors

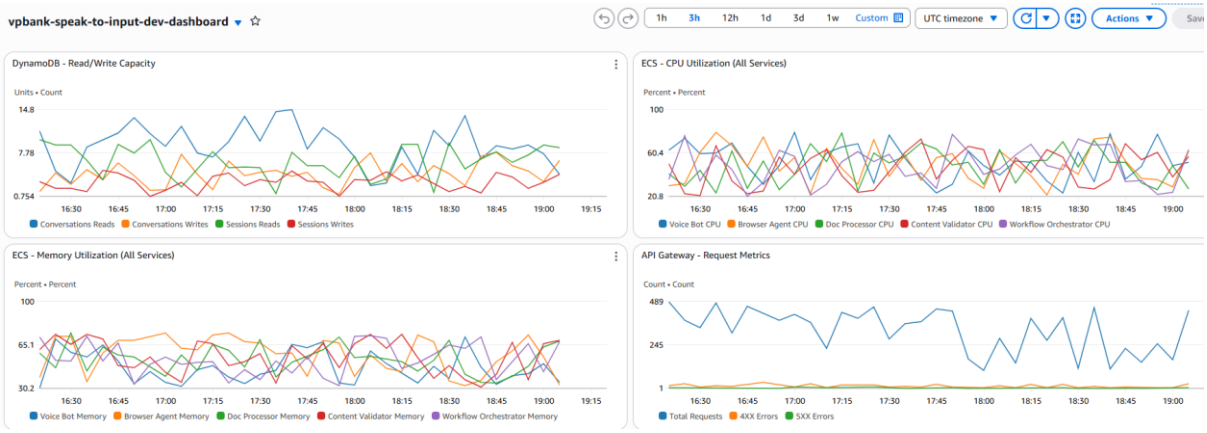


Figure <numb>: Monitoring infrastructure health

13.1.2 CloudWatch Alarms

Infrastructure Health Alarms: ECS task CPU/memory utilization, DynamoDB throttling, S3 request rates, API Gateway errors, notified to internal staffs if reaching the conditions

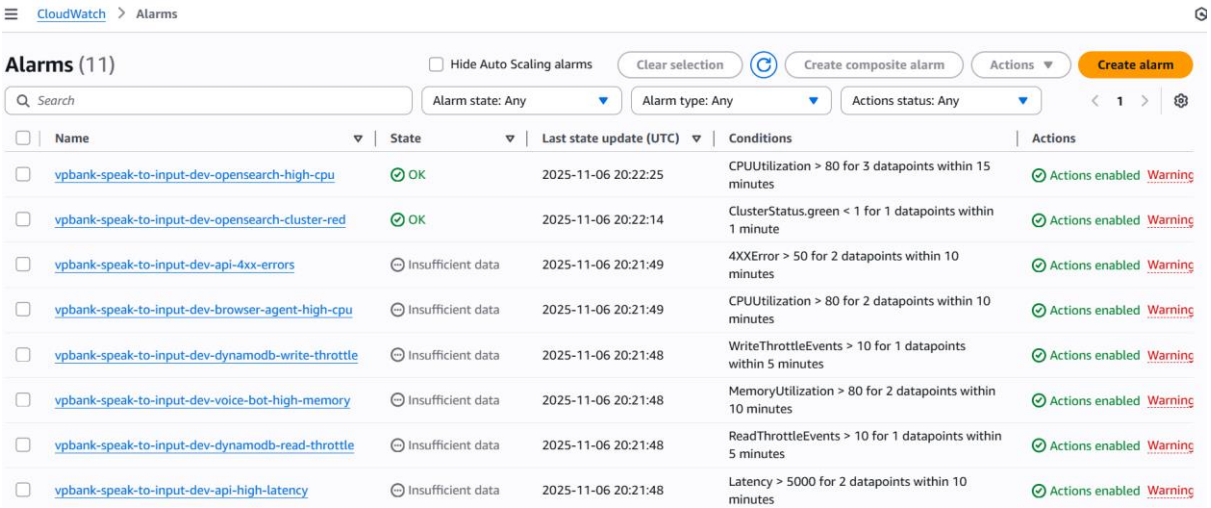


Figure <numb>: CloudWatch alarms to notify internal staff via email

13.2 Model Maintenance

13.2.1 PhoWhisper Model Fine-tuning

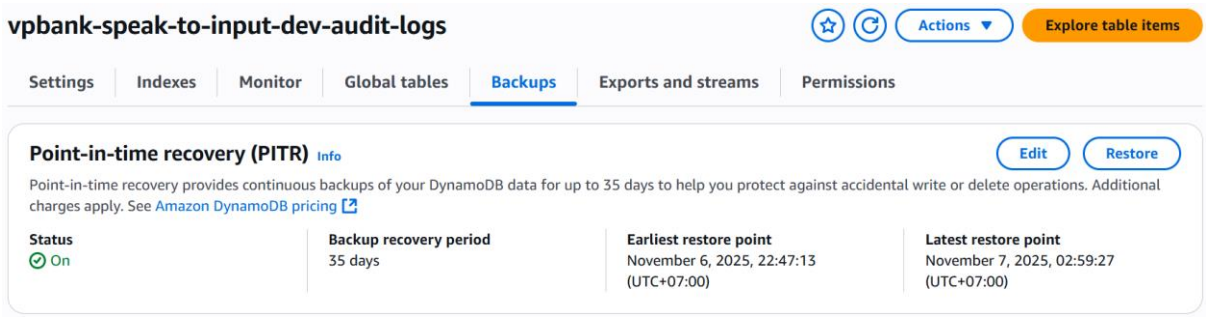
1. **Data Collection:** Monthly aggregation of anonymized production voice data
2. **Quality Assessment:** Evaluation of current model performance on new data
3. **Fine-Tuning:** Incremental training on SageMaker with updated dataset
4. **A/B Testing:** Gradual rollout (10% → 50% → 100%) comparing new vs. old model
5. **Model Promotion:** Promote to production if WER improvement >5% or no degradation
6. **Documentation:** Version tagging, performance metrics, training parameters logged

13.2.2 Intent Classification Model Updates

Bedrock Claude models are automatically updated by AWS. Custom prompt engineering and LangGraph workflows are versioned and tested before production deployment.

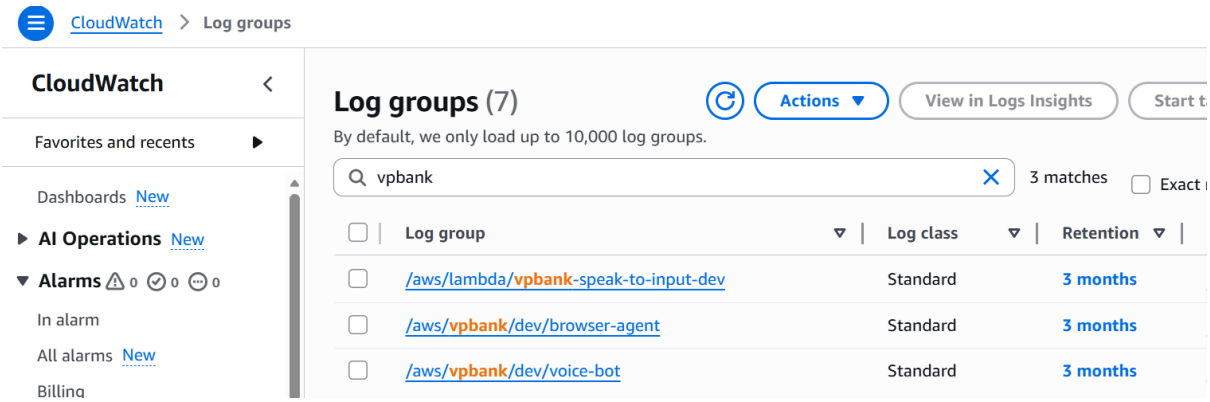
13.3 Database Maintenance

- **DynamoDB Backup:** Point-in-time recovery enabled, daily automated backups with 35-day retention



13.4 Log Management

- **Retention Policy:** CloudWatch Logs: 90 days, S3 Archive: 1 years


CloudWatch > Log groups
CloudWatch
Favorites and recents
Dashboards New
AI Operations New
Alarms 0 0 0
In alarm
All alarms New
Billing
Log groups (7)
Actions
View in Logs Insights
By default, we only load up to 10,000 log groups.
vpbank
3 matches
Exact
Log group
Log class
Retention
/aws/lambda/vpbank-speak-to-input-dev
Standard
3 months
/aws/vpbank/dev/browser-agent
Standard
3 months
/aws/vpbank/dev/voice-bot
Standard
3 months

14. Future Enhancements

This section outlines planned enhancements and future development roadmap for the VPBank Speak-to-Input system.

1. Sentiment Analysis

Real-time emotion detection to improve customer service quality and identify dissatisfaction

2. Multilingual Expansion

Support for regional Asian languages.

15. Project plan

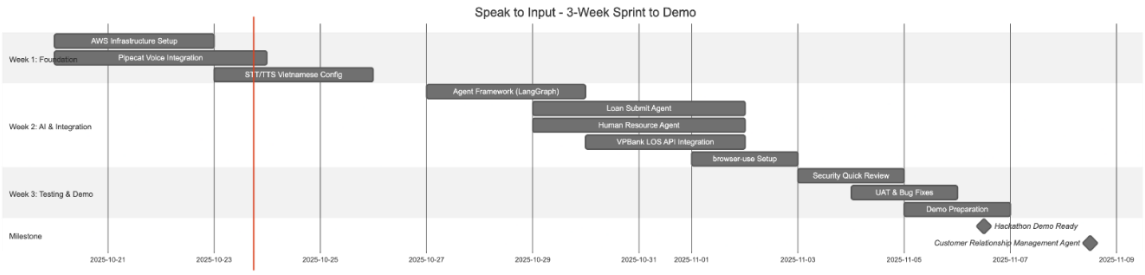


Figure 18 : Timeline Roadmap Project

Total Duration: 18 days (October 20 – November 6, 2025)

Week 1: Foundation & Core Setup (Oct 20–26)

- AWS infrastructure + Pipecat integration (parallel)
- Basic STT/TTS with Vietnamese support

Week 2: AI Agents & Integration (Oct 27 – Nov 2)

- Loan Submit Agent
- Human Resource (HR) Agent
- VPBank LOS API integration + browser–use setup

Week 3: Testing & Demo (Nov 3–6)

- Security review + UAT testing
- **Demo–ready prototype by Nov 6**

Key Overlaps: Testing begins while integration completes; Security review runs parallel to final integration work to save time.

Team Structure (5 Members)

Role/Function	Description	Team Size
Solution Architect	Technical Leadership	Hải Anh
AI/ML Engineer	LLM Integration, Agent Development, Pipecat Voice Agent	Hiếu Nghị

Fullstack Engineer	React Voice Widget, Backend API, System Integration	Minh Nghĩa
DevOps Engineer	AWS Infrastructure Implement	Đức Toàn
Project Manager	Tracking Progress, Quality Control	Lodi Bùi

Daily Task Distribution

Day	**Milestone**	**Owner**	**Deliverable**
1–2	AWS Infrastructure	Duc Toan	VPC, ECS cluster setup
3–4	Pipecat Integration	Nghia, Nghi	Voice gateway running
5–6	STT/TTS Working	Nghi	Vietnamese transcription demo
7–8	Agent Framework	Nghi	LangGraph orchestrator
9–10	Multi–Agent System	Nghia, Nghi	6 agents working
11–12	browser–use	Nghia	Browser automation live
13–14	VPBank Integration	Duc Toan	LOS/CRM connected
15–16	Security & Testing	Hai Anh	Test passed
17–18	UAT	AWS community members	User acceptance
19–20	Pilot Launch	All Team	20 users onboarded

16. Appendices

16.1 Glossary of Terms

- **AI Agent:** Autonomous software component that performs domain-specific tasks using artificial intelligence
- **ASR (Automatic Speech Recognition):** Technology that converts spoken language into written text
- **DFD (Data Flow Diagram):** Visual representation of data movement through a system
- **E2E (End-to-End):** Complete path from user input to system output
- **KPI (Key Performance Indicator):** Measurable value demonstrating system effectiveness
- **KYC (Know Your Customer):** Banking compliance process for customer identity verification
- **LLM (Large Language Model):** AI model trained on vast text data for language understanding
- **LOS (Loan Origination System):** Software managing loan application and approval process
- **RAG (Retrieval-Augmented Generation):** AI technique combining retrieval and generation for accurate responses
- **ROI (Return on Investment):** Performance measure evaluating efficiency of an investment
- **STT (Speech-to-Text):** Conversion of spoken words into written text
- **TTS (Text-to-Speech):** Conversion of written text into spoken audio
- **VAD (Voice Activity Detection):** Technology identifying presence of human speech in audio
- **WER (Word Error Rate):** Metric measuring speech recognition accuracy
- **WebRTC:** Real-time communication technology for audio/video transmission in browsers

16.3 Sample API Requests & Responses

Complete API collection available in Postman format at:

https://github.com/vpbank/speak-to-input-api-collection

16.4 Test Scenarios & Scripts

Comprehensive test suite including:

- Unit tests for voice pipeline components

- Integration tests for end-to-end workflows
- Performance test scripts for load and stress testing
- Security test cases for penetration testing
- UAT test scenarios with expected outcomes

16.5 Validation Results

Initial validation results from pilot testing:

Metric	Target	Actual Result
Word Error Rate (Vietnamese)		
Workflow Correction Rate		
E2E Voice Latency (P95)		
Intent Classification Accuracy		
User Satisfaction Score		
System Availability		

15.6 References

— End of Technical Documentation —

[1] McKinsey & Company. (2017). Automating the bank's back office. McKinsey Digital. Retrieved from https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/automating-the-banks-back-office; The Lab Consulting. (2016). Robotics in Banking with RPA Use Case Examples. Retrieved from https://thelabconsulting.com/robotics-in-banking-with-4-rpa-use-case-examples/

[2] Panko, R.R. (2008). What We Know About Spreadsheet Errors. Journal of End User Computing; DocuClipper. (2025). 67 Data Entry Statistics For 2025. Retrieved from https://www.docuclipper.com/blog/data-entry-statistics/

[3] Mastercard. (2023). Financial Reconciliation Case Study. Technology Review; DocuClipper. (2025). Bank Statement Processing Accuracy Statistics. Retrieved from https://www.docuclipper.com/blog/data-entry-statistics/

[4] McKinsey & Company. (2017). The transformative power of automation in banking. McKinsey Financial Services; Accenture. (2019). Banking Automation Research. Retrieved from https://www.accenture.com/us-en/industries/banking/commercial-corporate-banking; UiPath. (2022). BankDhofar Automation Case Study. Retrieved from https://www.uipath.com/resources/automation-case-studies/bankdhofar-sets-standards-for-automation

[5] AWS. (2025). Amazon Bedrock Pricing. Retrieved from https://aws.amazon.com/bedrock/pricing/; Anthropic. (2025). Claude API Pricing Documentation. Retrieved from https://www.anthropic.com/pricing

[6] General Statistics Office of Vietnam, & UNFPA. (2021). Population ageing and older persons in Viet Nam: Key findings from the 2019 Census. Hanoi, Vietnam: UNFPA. Retrieved from https://vietnam.unfpa.org/sites/default/files/pub-pdf/ageing_report_from_census_2019_eng_final27082021.pdf

[7] UNICEF Viet Nam, & General Statistics Office of Viet Nam. (2016). Results of the survey on people with disabilities in Viet Nam. Hanoi, Vietnam: UNICEF. Retrieved from https://www.unicef.org/vietnam/media/2786/file/Main%20report%20people%20with%20disabilities%20survey.pdf

[8] Zuko Analytics. (2024). Form benchmark report 2024. Zuko Analytics Ltd. Retrieved from https://www.zuko.io/benchmarking/industry-benchmarking