📌 EPIC 1: AI-Powered Smart Email Generation Engine (MVP)

Epic Description

Build an AI-powered multilingual assistant that detects user intent (leave, WFH, resignation, etc.), detects language (English/Hindi/Hinglish), collects missing required information dynamically, generates a professional email, and integrates with the native email client for sending.

This MVP will support the 9 highest-frequency employee email scenarios.

🥇 USER STORY 1 – Intent Detection & Language Detection Engine (Core Foundation)

Backend – AI/LLM Team

User Story 1.1

As a system, I want to detect the user's intent from free-text input, so that the correct email template workflow is triggered.

In-Scope

- Intent classification for:
    - Sick Leave
    - Casual Leave
    - WFH
    - Half Day
    - Emergency Leave
    - Late Coming
    - Meeting Request
    - Weekly Report
    - Resignation
- Confidence scoring
- Fallback intent

Out-of-Scope

- Advanced HR policy validation
- Multi-intent chaining

Acceptance Criteria

- AC1: System detects correct intent with ≥ 90% accuracy on test dataset
- AC2: Confidence score returned
- AC3: If confidence < 70%, trigger clarification flow

Validation Rules

- Input length: 3–500 characters
- No empty input
- Unicode support required

Error Messages

- “Please enter your request.”
- “I couldn’t understand your request. Can you rephrase?”

DoD

- Model trained & validated
- Test dataset coverage >500 examples
- API documented
- Unit tests >85% coverage

User Story 1.2

As a system, I want to detect the language (English/Hindi/Hinglish), so that the email tone matches user language.

In-Scope

- Language classification (EN / HI / HINGLISH)
- Script detection (Devanagari vs Latin)

Out-of-Scope

- Translation engine

Acceptance Criteria

- AC1: Correct language classification ≥ 92%

- AC2: Hinglish detection supported

Validation Rules

- Mixed-script handling supported

Error Message

- None (silent fallback to English)

DoD

- Language classifier deployed
- Performance benchmark documented

🥈 USER STORY 2 – Dynamic Information Collection Engine

Backend – AI + Orchestration Team

User Story 2.1

As a system, I want to ask follow-up questions if required fields are missing, so that email content is complete.

Required Fields Logic (Example: Sick Leave)

- Leave Date
- Duration
- Reason (optional for professional tone)

In-Scope

- Slot-filling logic
- Context memory (session-based)
- Structured JSON output

Out-of-Scope

- Persistent long-term memory

Acceptance Criteria

- AC1: Missing fields trigger question
- AC2: Context retained across conversation
- AC3: JSON structure returned before generation

Validation Rules

- Date format: YYYY-MM-DD
- Max reason length: 250 characters

Error Messages

- “Please provide leave date.”
- “Invalid date format. Use DD-MM-YYYY.”

DoD

- Slot schema defined for all 9 intents
- Validation rules implemented
- QA conversation flows tested

🥉 USER STORY 3 – Email Generation Engine

LLM Team

User Story 3.1

As an employee, I want a professionally formatted email generated automatically, so that I can send it without editing.

In-Scope

- Tone adjustment (formal / semi-formal)
- Language-aware generation
- Manager name insertion
- Subject line auto-generation
- Structured output:
    - Subject
    - Greeting
    - Body
    - Closing

Out-of-Scope

- Legal compliance customization per company

Acceptance Criteria

- AC1: Email contains subject
- AC2: Correct manager name inserted
- AC3: Language consistent with user language
- AC4: No hallucinated data

Validation Rules

- Subject length: 5–120 characters
- Email body: 50–400 words
- No placeholder text allowed

Error Messages

- “Unable to generate email. Please try again.”

DoD

- Prompt templates documented
- Guardrails implemented
- Toxicity filtering enabled
- QA reviewed 50 test samples per intent

🏗️ USER STORY 4 – Native Email App Integration

Backend + Mobile Team

User Story 4.1

As an employee, I want the generated email to open in my native email app, so that I can send it instantly.

In-Scope

- Pre-fill:
    - To (Manager email)

    - Subject
    - Body
- Android & iOS support

Out-of-Scope

- Direct sending without user confirmation

Acceptance Criteria

- AC1: Opens default email app
- AC2: Fields auto-filled correctly
- AC3: No formatting breaks

Validation Rules

- Valid email regex
- Subject must not be empty

Error Messages

- “Manager email not found.”
- “No email app detected.”

DoD

- Deep linking tested
- QA on Android & iOS
- No PII stored in logs

🎨 USER STORY 5 – Frontend Interaction Layer

Frontend Team

User Story 5.1 – Input Screen

As a user, I want a simple text input interface, so that I can type my request naturally.

In-Scope

- Text input box
- Send button

- Loading indicator

Out-of-Scope

- Voice input (Phase 2)

Acceptance Criteria

- AC1: Submit triggers intent API
- AC2: Loader displayed during processing
- AC3: Error message shown if API fails

Validation Rules

- Max input: 500 characters
- No blank submission

Error Messages

- “Please type your request.”

DoD

- Responsive UI
- API connected
- UI test cases passed

User Story 5.2 – Clarification UI

As a user, I want the app to ask me missing details conversationally, so that I can complete my request easily.

In-Scope

- Chat-style UI
- Dynamic follow-up questions
- Editable responses

Out-of-Scope

- Multi-threaded conversation history

Acceptance Criteria

- AC1: Questions appear sequentially
- AC2: User responses stored in session

Validation Rules

- Date picker integration preferred

Error Messages

- “This field is required.”

DoD

- Session handling implemented
- UX tested

User Story 5.3 – Email Preview Screen

As a user, I want to preview the generated email before sending, so that I can edit if needed.

In-Scope

- Editable email preview
- Copy option
- Send button

Out-of-Scope

- Rich HTML formatting

Acceptance Criteria

- AC1: Subject editable
- AC2: Body editable
- AC3: Send opens email client

Validation Rules

- Subject cannot be empty

Error Message

- “Subject is required.”

DoD

- Editing functional
- No formatting breaks

📊 USER STORY 6 – AI Analytics & Monitoring

AI Analytics Team

User Story 6.1

As a product team, I want analytics on intent frequency and model confidence, so that we can improve accuracy.

In-Scope

- Intent frequency tracking
- Language distribution
- Confidence distribution
- Drop-off rate tracking

Out-of-Scope

- Employee performance tracking

Acceptance Criteria

- AC1: Dashboard available
- AC2: Daily summary generated
- AC3: Low-confidence alerts (<60%)

Validation Rules

- No raw PII stored
- Aggregated data only

DoD

- Dashboard deployed
- Privacy reviewed
- Logging policy approved

🔢 Implementation Sequence (Strict Order)

1. Intent Detection Engine
2. Language Detection
3. Slot Filling & Validation
4. Email Generation Prompt Templates
5. Email Preview UI
6. Native Email Integration
7. Analytics Dashboard

🏁 MVP Definition of Done (Global)

- All 9 intents supported
- Multilingual detection working
- 90%+ classification accuracy
- 95% successful email generation rate
- Native email integration tested
- No critical security issues
- Load tested for 10K users
- Documentation completed
- QA sign-off obtained