Planeación de sistemas de software

Gpo 101

Sample User Acceptance Test Plan

HOWLX

Version 1.0

Test Plan Number: 001

Alejandra Coeto Sánchez

Diego de Jesús Esparza Ruiz

Jesús Adrián López Gaona

Luis Gerardo Juárez García

Mónica Soberón Zubía

Santiago De la Riva Juárez

Monterrey, NL.
10 de Junio, 2025

Table of Contents

1. Introduction

1.1 Test Objectives

The acceptance test of the HowlX system should validate from both the business and technical perspective that:

- AI-powered call analysis processes work correctly, including transcription, sentiment analysis, and automated report generation
- The system is intuitive and user-friendly as measured by user surveys and usability testing
- Role-based access control (RBAC) functions properly for different user types (administrators, supervisors, consultants)
- RAG-based intelligent chat provides accurate responses with proper source citations from call transcriptions
- Smart recommendations generate actionable insights based on historical call data
- Data security and privacy controls are effective to prevent unauthorized access to sensitive call data
- All AI analyses and calculations produce accurate results as defined by business rules

The objective of acceptance testing is to validate system operation and usability across different user roles and business scenarios. At the conclusion of acceptance testing, end-users will have high confidence that HowlX meets their customer service analysis and improvement needs.

1.2 Scope of Testing

The acceptance test of the HowlX system will include:

- Call processing workflow: Audio upload, transcription, and AI analysis
- User management: Company registration, client management, and role assignments
- AI features: RAG chat, smart recommendations, sentiment analysis
- Analytics dashboards: Performance metrics and reporting
- Form integration: Google Forms connectivity for customer feedback
- Administrative functions: User role management and system configuration

The acceptance test will **not** include:

- Direct integrations with specific video calling platforms (will use simulated data)
- Real-time call processing during live calls
- Advanced system administration functions beyond basic user management

1.3 System Overview

HowlX is a comprehensive AI-powered platform that analyzes customer service calls to generate actionable business insights. The system accepts audio files from various sources, processes them through advanced AI models, and provides users with:

- Automated transcription using state-of-the-art speech recognition
- Intelligent analysis including sentiment detection, topic extraction, and risk identification
- Interactive dashboards showing performance metrics and trends
- RAG-powered chat interface for querying historical call data
- Smart recommendations for improving customer service quality
- Multi-tenant architecture supporting multiple companies and clients

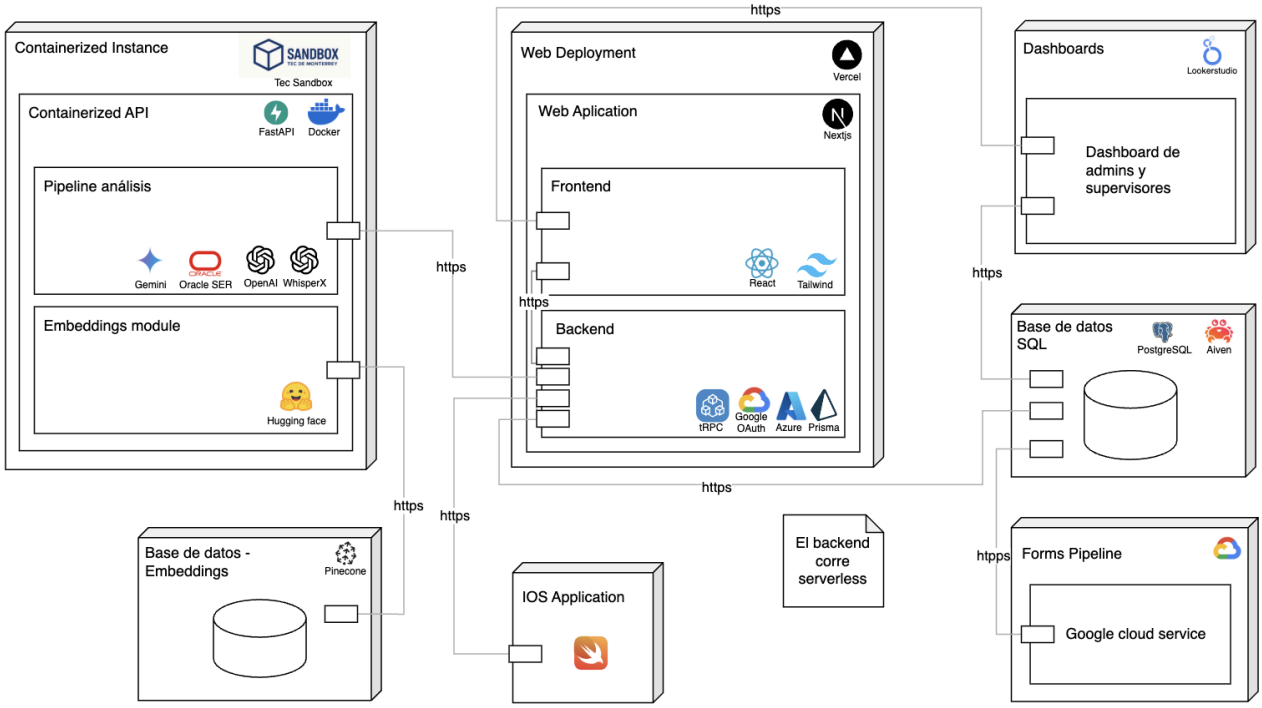

Image 1: Architecture diagram.

1.4 Definitions/Acronyms

Acceptance Testing
Testing that ensures HowlX will work in real customer service environments to meet business needs based on pre-defined acceptance criteria.

RAG (Retrieval-Augmented Generation):
AI technique combining information retrieval with text generation to provide contextual responses with source citations.

Smart Recommendations
AI-generated suggestions for improving customer service based on call analysis patterns.

RBAC (Role-Based Access Control)
Security model that restricts system access based on user roles (administrator, supervisor, consultant).

1.5 References

- Requirements Specification Document for the XYZ System
- Test Standards
- Test Procedures
- Test Plan Notebook
- Payroll Policy and Procedures Notebook

2. Approach

2.1.1 Constraints

- Two-week testing window may be insufficient for comprehensive testing of all AI features and edge cases
- AI model dependencies on external APIs (OpenAI, Gemini) may introduce variability in results
- Limited real call data available for testing due to privacy constraints
- Multi-model AI testing requires validation across different AI providers
- Limited available budget for LLM testing

2.2 Coverage

Test coverage will be measured by:

- Feature coverage: Percentage of user stories and acceptance criteria validated
- Role coverage: Testing across all user roles (admin, supervisor, consultant)
- AI model coverage: Validation with different AI providers and configurations
- Data flow coverage: End-to-end validation of call processing workflows

2.2.1 Software Components

The following critical software modules will be tested:

- Frontend web application (React-based interface)
- Backend API services (Python/Node.js)
- AI processing pipeline (transcription, analysis, RAG)
- Database layer (relational database with optimized queries)
- Authentication and authorization (RBAC implementation)
- IOS application (Swift implementation)

2.2.2 User Stories / epics

All criticall user stories / epics are:

- User Management and Authentication - Company registration, user roles (admin, supervisor, consultant), and RBAC implementation
- Call Processing Workflow - Audio upload, transcription, and AI analysis pipeline
- Intelligent AI Features - RAG-based chat, sentiment analysis, topic extraction, and risk identification
- Data Analytics - Performance metrics, reporting dashboards, and trend analysis
- AI Model Modularity - Multi-provider AI integration (OpenAI, Gemini) and fallback mechanisms

2.3 Test Tools

- Cypress: End-to-end automated testing
- Pytest: Backend unit and integration testing
- Manual testing: User experience validation
- Postman: API testing and validation
- GitHub Actions: Automated test execution
- Defect log system: For acceptance testing

2.4 Test Type

The following types of testing will be performed during acceptance testing:

- Functional testing: Core business process validation
- Usability testing: User experience evaluation across different roles
- Security testing: RBAC and data privacy validation
- Performance testing: System response times and scalability
- Regression testing: Ensuring new features don't break existing functionality

2.5 Test Data

- Mock call transcriptions: Structured test data simulating real customer interactions
- User profile data: Test accounts for different roles and companies
- AI model responses: Controlled test data for validating AI analysis features
- Form submissions: Sample customer feedback data for integration testing

3. Plan

3.1 Test Team

The following people will be on the acceptance test team:

Location	Name	Level of involvement	Responsibilities
Monterrey	Alejandra Coeto	40 hrs/wk	Design test cases for call analysis.
Monterrey	Santiago de la Riva	40 hrs/wk	Design and execute test cases, create test data, write test summary report
Monterrey	Jesus Lopez	40 hrs/wk	Design and execute test cases for the RAGChat, AI Modularity and audio processing pipeline
Monterrey	Luis Juarez	40 hrs/wk	Design and execute test cases for the Logs and Role Management Component (User Management)
Monterrey	Monica Soberon	40 hrs/wk	Design and execute test cases, build employee test tables
Monterrey	Diego Esparza	40 hrs/wk	Design and execute test cases, create test data.

3.2 Environmental Needs

3.2.1 Test Environment

Hardware

- Desktop/laptop computers with modern browsers (Chrome, Firefox, Safari)
- Minimum specifications: Intel i5 processor, 8GB RAM, 128GB storage
- Network connectivity for API testing and AI model integration
- Audio capability for testing file uploads

Software

- HowlX web application (staging environment)
- Chrome Web Browser
- Audio file samples for testing transcription (neorisTests.wav)
- Access to AI model APIs (Gemini)

3.2.2 Test Lab

- Dedicated testing environment isolated from production
- Mock data services for AI models to ensure consistent results
- Database with test data including companies, users, and call records
- Monitoring tools for performance and error tracking

3.3 Training

Acceptance testers will receive training on:

- HowlX platform overview and business objectives
- User role scenarios and typical workflows
- AI feature validation techniques
- Test case execution and defect reporting procedures

4. Features to be Tested

4.1 User Stories / epics (story name and most important acceptance criteria)

4.1.1 User Management and Authentication

US25 - Registro de admins, supervisores y empleados

- **AC1**. El usuario debe acceder con una cuenta con permiso de administrador
- **AC2.** El usuario puede cambiar el rol entre administrador, supervisor o consultor
- **AC3**. El usuario podrá gestionar a las personas supervisadas por otro usuario con rol de supervisor
- **AC4**. El usuario puede eliminar el rol a cualquier usuario

US16 - Registro de empresas y clientes

- **AC1.** El usuario puede crear una nueva empresa
- **AC2.** El usuario puede crear un nuevo cliente (que pertenece a una empresa)
- **AC3.** El usuario puede modificar empresas existentes
- **AC4.** El usuario puede modificar clientes existentes
- **AC5.** El usuario puede eliminar empresas existentes
- **AC6.** El usuario puede eliminar clientes existentes
- **AC7.** El usuario puede ver y buscar la lista de empresas
- **AC8.** El usuario puede ver, filtrar y buscar la lista de clientes

4.1.2 Call Processing Workflow

US8 - Calificación de Eficiencia de Llamadas

- **AC1.** El usuario existe en el sistema
- **AC2.** La cuenta del usuario debe tener un perfil de tipo gerente
- **AC3.** La llamada o el texto tiene que ser subida previamente a la aplicación
- **AC4.** La duración del input de la llamada tiene que ser mayor a 2 minutos y en el caso de introducir un texto su largo tiene que ser mayor a 500 palabras
- **AC5.** En caso de haberlos los parámetros pre definidos tienen que ser seleccionados o escritos por el usuario para la evaluación

4.1.3 Intelligent AI Features

US17 - Smart recommendations

- **AC1.** El usuario puede entrar a la sección de Smart Recommendations desde el sistema
- **AC2.** El usuario puede elegir entre distintos tipos de recomendaciones (por cliente, por agente, tareas semanales, etc.)
- **AC3.** El sistema identifica automáticamente el tipo de recomendación seleccionada
- **AC4.** El sistema muestra la información correspondiente sin necesidad de recargar la página
- **AC5.** El usuario puede cambiar entre tipos de recomendación
- **AC6.** Cada tipo de recomendación muestra resultados claros, accionables y basados en llamadas reales
- **AC7.** El diseño permite navegar de forma sencilla entre los módulos disponibles

US14 - Búsqueda Avanzada en Transcripciones RAG

- **AC1.** El usuario podrá acceder a un chatbot integrado en la página, visible y disponible desde la interfaz principal de la aplicación
- **AC2.** El usuario podrá seleccionar llamadas o transcripciones relevantes para agregarlas como contexto adicional en sus consultas al chatbot
- **AC3.** El usuario podrá realizar preguntas relacionadas con el contenido de las llamadas seleccionadas, y el chatbot responderá con información relevante basada en esos datos
- **AC4.** El usuario podrá visualizar, junto con cada respuesta generada por el chatbot, una cita que indique claramente la fuente de la información, así como una estimación de la probabilidad o nivel de relevancia de dicha fuente

US5 - Consulta Contextual de Conversaciones (Chat general)

- **AC1.** La plataforma identifica la llamada o llamadas relevantes al hacer mi pregunta
- **AC2.** La plataforma me permite redactar una pregunta
- **AC3.** La plataforma permite utilizar la inteligencia artificial para generar la respuesta
- **AC4.** Cuando lleno mi mensaje veo alguna confirmación que el mensaje a sido recibido por la IA y está generando una respuesta: ver el status de mi pregunta
- **AC5.** Cuando recibo el mensaje utiliza correctamente el contexto de la llamada deseada
- **AC6.** Recibo un mensaje y es desplegado correctamente
- **AC7.** La plataforma de deja saber si hubo algún error o algo salió mal

4.1.4 Data Analytics

US10 - Historial Integral de Llamadas (Logs)

- **AC1.** El usuario existe en el sistema
- **AC2.** La cuenta del usuario debe tener un perfil de cualquier tipo
- **AC3.** La llamada o el texto tiene que ser subida previamente a la aplicación
- **AC4.** El usuario tiene que haber atendido o participado previamente en la llamada

4.1.5 AI model modularity

US16 - AI modularity

- **AC1.** Entrada libre: la UI permite guardar modelo y API-key propios
- **AC2.** Asignación por función: el usuario puede elegir un modelo distinto para chat, transcripción y reportes
- **AC3.** Validación de capacidad: si el modelo seleccionado no soporta la función (p. ej. audio), la opción se bloquea con un aviso
- **AC4.** Aplicación inmediata: los cambios se reflejan en /health sin reiniciar el servidor
- **AC5.** Almacenamiento seguro: las credenciales se guardan cifradas y pueden editarse o borrarse desde la UI
- **AC6.** Proveedor "custom": requiere base_url y debe pasar la prueba de conexión; si falla, se revierte al proveedor por defecto

5. Features Not to be Tested

5.1 Out of Scope for Acceptance Testing

- Real-time call recording during live video conferences
- Advanced system administration functions (server management, database optimization)
- Integration with specific video platforms beyond file upload capability
- Advanced AI model training or customization
- Scalability testing beyond normal usage patterns
- Mobile application
- Custom AI model
-

6. Testing Procedures

6.1 Test Execution

6.1.1 Test Cases

For each business process to be tested, the acceptance tester will execute a set of pre-defined test cases. Each test case will have a series of actions and expected results. As each action is performed, the results are evaluated. If the observed results are equal to the expected results, a checkmark is placed in the “pass” column. If the observed results are not equal to the expected results, a checkmark is placed in the “fail” column.

6.1.2 Order of Execution

1. System Setup and Authentication
    a. User registration and login
    b. Role assignment and verification

2. Data Management
    a. Company creation
    b. Client registration
    c. User role management

3. Core AI Features

    a. Call upload and transcription
    b. AI analysis and report generation
    c. RAG chat functionality

4. Advanced Features

    a. Smart recommendations
    b. Analytics dashboards
    c. Form integrations

5. Integration and Performance
    a. API endpoint validation
    b. Performance under load
    c. Security testing

6.1.3 Test Data

To perform acceptance testing, test data will include:

- Standardized audio files for consistent transcription testing
- Mock user accounts representing different organizational roles
- Sample call scenarios covering various customer service situations

6.2 Pass/Fail Criteria

- 70% of user epics meet all acceptance criteria
- AI features provide accurate results within minimal hallucinations (less than 15%)
- User experience meets usability standards across all roles
- No critical or high-severity defects remain unresolved

7. Risks and Contingencies

7.1 High-Risk Areas

7.1.1 AI Model Dependencies

- **Risk Level**: High
- **Impact**: Core functionality depends on external AI services
- **Mitigation**: Implement fallback models and error handling
- **Contingency**: Manual verification processes if AI fails

7.1.2 Data Privacy and Security

- **Risk Level**: High
- **Impact**: Customer call data requires strict privacy protection
- **Mitigation**: Comprehensive security testing and access controls
- **Contingency**: Immediate security audit if vulnerabilities found

7.2 Medium-Risk Areas

7.2.1 User Experience Complexity

- **Risk Level**: Medium
- **Impact**: Complex AI features may confuse users
- **Mitigation**: Extensive usability testing and user training
- **Contingency**: UI simplification and enhanced documentation

7.1.3 Performance Under Load

- **Risk Level**: Medium
- **Impact**: System may slow down with large datasets
- **Mitigation**: Performance testing with realistic data volumes
- **Contingency**: Optimization strategies and infrastructure scaling

7.2.2 Integration Reliability

- **Risk Level**: Medium
- **Impact**: External service failures could impact functionality
- **Mitigation**: Robust error handling and retry mechanisms
- **Contingency**: Graceful degradation when services unavailable