

Hackathon Project Phases Template

Project Title:

Audio transcription app using OpenAi Whisper

Team Name:

EchoScribe

Team Members:

- S. Srinija
 - K. Sidhartha Rao
-

Phase-1: Brainstorming & Ideation

Objective:

Develop an AI-powered audio transcription tool for accurate and efficient speech-to-text conversion.

Key Points:

1. Problem Statement:

- Many users struggle with transcribing audio recordings accurately and efficiently.
- Existing transcription tools may have limitations in handling different accents, languages, and background noise.
- Users need a reliable, fast, and cost-effective solution for converting speech to text.

2. Proposed Solution:

- An AI-powered transcription application leveraging **OpenAI Whisper** for high-accuracy speech recognition.
- Support for multiple languages and accents to improve accessibility.
- Options for real-time transcription and batch processing of recorded audio.
- User-friendly interface for uploading, processing, and downloading transcriptions.

3. Target Users:

- **Students & Researchers:** Transcribing lectures and interviews.
- **Journalists & Content Creators:** Converting interviews and podcasts into text.
- **Business Professionals:** Generating meeting notes and summaries.
- **Individuals with Accessibility Needs:** Assisting those who prefer text-based communication.

4. Expected Outcome:

- A **functional, accurate, and efficient** transcription tool powered by OpenAI Whisper.
 - Seamless **speech-to-text conversion** with options for text formatting and exporting.
 - User satisfaction through a **smooth and intuitive UI experience**.
-

Phase-2: Requirement Analysis

Objective:

Define the technical and functional requirements for the Transcription App.

Key Points:

1. Technical Requirements:

- Programming Language: **Python**
- Backend: **Flask, Hugging Face**
- Frontend: **HTML, CSS**
- Database: **Sqlite3**

2. Functional Requirements:

- Allow users to upload audio files for transcription.
- Process and convert audio into text using OpenAI Whisper.
- Provide options for different transcription models (e.g., base, large).
- Enables copying the transcribed text.

3. Constraints & Challenges:

- Ensuring enough computation power for using the whisper large with a size of 1.85B

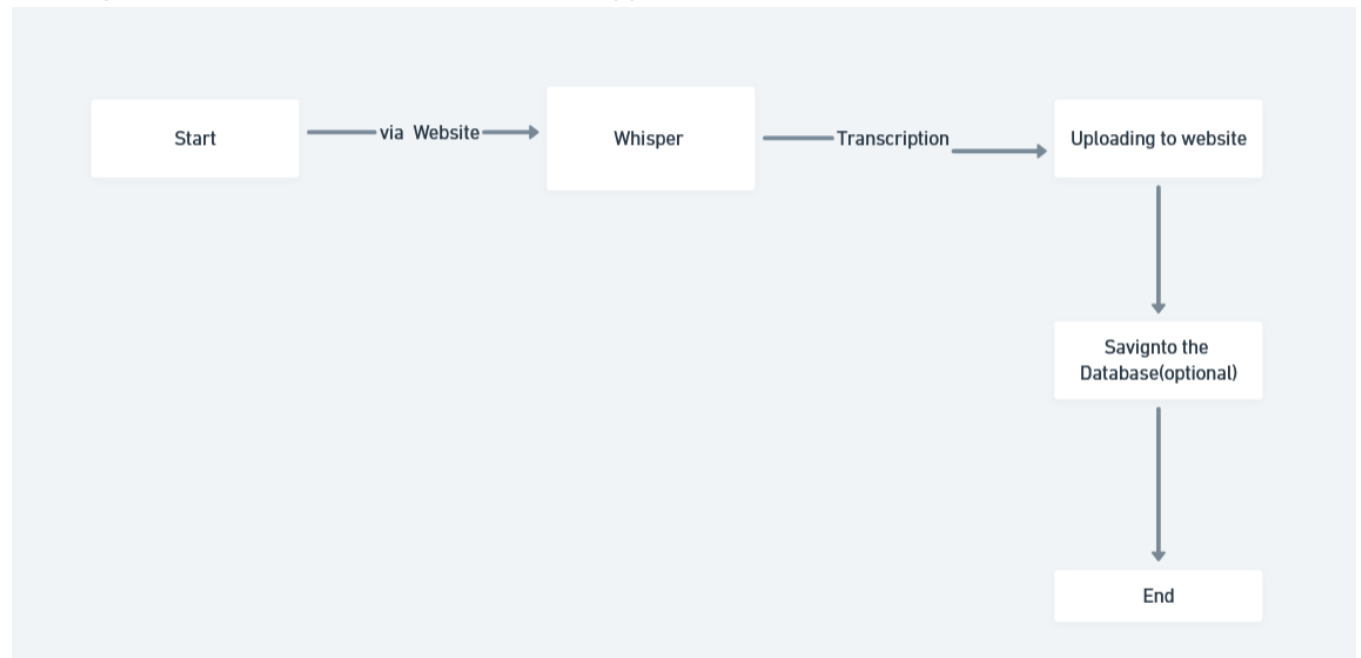
parameter.

- 0 Providing a **smooth UI experience** with HTML.

Phase-3: Project Design

Objective:

Develop the architecture and user flow of the application.



Key Points:

1. System Architecture:

- User uploads an audio file or records audio via the UI.
- The request is processed using **OpenAI Whisper API** for transcription.
- The AI model transcribes the speech into text with high accuracy.
- The frontend displays the transcribed text, with options to edit, copy, or download.

2. User Flow:

- **Step 1:** User uploads an audio file or records live audio.
- **Step 2:** The backend processes the file using **OpenAI Whisper** for transcription.
- **Step 3:** The transcribed text is displayed in a clean and readable format.
- **Step 4:** Users can edit, format, or download the text as needed.

3. UI/UX Considerations:

- **Minimalist, user-friendly interface for seamless navigation.**
 - **Progress indicator while transcription is in process.**
 - **Support for multiple languages and different accents.**
 - **Dark & light mode for a better user experience.**
 - **Download options (TXT, DOCX, or PDF).**
-

Phase-4: Project Planning (Agile Methodologies)

Objective:

Break down development tasks for efficient completion.

Sprint	Task	Priority	Duration	Deadline	Assigned To	Dependencies	Expected Outcome
Sprint 1	Environment Setup & API Integration	🔴 High	6 hours (Day 1)	End of Day 1	Srinija	Python, model setup	Model is loaded & working
Sprint 1	Frontend UI Development	🟡 Medium	2 hours (Day 1)	End of Day 1	Sidhartha	Response format finalized	Basic UI with input fields
Sprint 2	Vehicle Search & Comparison	🔴 High	3 hours (Day 2)	Mid-Day 2	Sidhartha	UI elements ready	Search functionality with filters
Sprint 2	Error Handling & Debugging	🔴 High	1.5 hours (Day 2)	Mid-Day 2	Srinija	Model logs	Improved Model stability
Sprint 3	Testing & UI Enhancements	🟡 Medium	1.5 hours (Day 2)	Mid-Day 2	Entire Team	UI layout completed	Responsive UI, better user experience
Sprint 3	Final Presentation & Deployment	🟢 Low	1 hour (Day 2)	End of Day 2	Entire Team	Working prototype	Demo-ready project

Sprint Planning with Priorities

Sprint 1 – Setup & Integration (Day 1)

(🔴 High Priority) Set up the **environment** & install dependencies.

(🔴 High Priority) Integrate **OpenAI Whisper**.

(🟡 Medium Priority) Build a **basic UI with input fields**.

Sprint 2 – Core Features & Debugging (Day 2)

(🔴 High Priority) Implement **search & comparison functionalities**. (🔴

High Priority) Debug API issues & handle **errors in queries**.

Sprint 3 – Testing, Enhancements & Submission (Day 2)

(🟡 Medium Priority) Test API responses, refine UI, & fix UI bugs.

(🟢 Low Priority) Final **demo preparation & deployment**.

Phase-5: Project Development

Objective:

Implement core features of the Trancription App.

Key Points:

- 1. **Technology Stack Used:**
 - **Frontend:** HTML, CSS
 - **Backend:** Transformers,Flask
 - **Programming Language:** Python
- 2. **Development Process:**
 - Implement **API key authentication** and integrate **OpenAI Whisper API** for transcription.
 - Develop **audio processing logic** to handle different formats (MP3, WAV, etc.).
 - Optimize transcription **accuracy and speed** using appropriate Whisper models.
 - Implement **text formatting options** (e.g., punctuation, paragraph structuring).
- 3. **Challenges & Fixes:**
 - **Challenge:** Large file sizes may slow down processing.
Fix: Implement file size limits and allow background processing.
 - **Challenge:** Handling different accents and background noise.
Fix: Use **Whisper’s advanced models** and allow **user feedback** to improve results.

Phase-6: Functional & Performance Testing

Objective:

Ensure that the AutoSage App works as expected.

Test Case ID	Category	Test Scenario	Expected Outcome	Status	Tester
TC-001	Functional Testing	Input of the Audio file	The Transcription of the audio	✔ Passed	Srinija

TC-002	Functional Testing	Audio with background noise.	Output is perfect a the audio given.	✓ Passed	Sidhartha
TC-003	Performance Testing	Response time is based on the model if we use the large model it higher	Model should be pipelined for getting results quickly.	⚠ Needs Optimization	Sidhartha
TC-004	Bug Fixes & Improvements	Fixed incorrect response.	Data accuracy should be improved.	✓ Fixed	Srinija
TC-005	Final Validation	Ensure UI is responsive across devices.	UI should work on mobile & desktop.	✓ Passed	Sidhartha
TC-006	Deployment Testing	Not yet deployed	Website should be deployed	Not Deployed	Srinija

Final Submission

1. **Project Report Based on the templates**
2. **Demo Video (3-5 Minutes)**
3. **GitHub/Code Repository Link**
4. **Presentation**