Sprint 1 Plan
Product Name: HuddleAI
Sprint Completion Date: 5/9
Revision Number & Date: Revision 1, 5/2

———

Goal

Establish the core upload-to-transcript pipeline for HuddleAI by implementing file upload functionality, cloud storage integration, and basic transcription using a selected speech-to-text engine.

———

Task Listing (Organized by User Story)

User Story 1

As a user, I want to upload a meeting recording (audio/video) so that HuddleAI can process it for transcription.

- Set up backend API to accept file uploads (3 hrs)
- Implement file validation (type, size checks) (2 hrs)
- Integrate frontend file upload UI (React/Next.js) (3 hrs)
- Store metadata in database (name, timestamp, etc.) (2 hrs)
- Configure cloud storage bucket (Firebase Storage or AWS S3) (2 hrs)
- Connect backend to upload received files to cloud storage (3 hrs)

Total for User Story 1: 15 hours

User Story 2

As a user, I want to see a raw transcript of the meeting to verify what was said.

- Compare Whisper vs. Google STT (accuracy, latency, pricing) (2 hrs)
- Evaluate diarization tool options (e.g., pyannote.audio) (2 hrs)
- Integrate selected STT tool into backend pipeline (4 hrs)
- Build transcription interface to convert audio into text (3 hrs)
- Format and export transcript as JSON and plain text (2 hrs)

Total for User Story 2: 13 hours

User Story 3

As a developer, I want to verify the pipeline works with test files and is ready for further feature expansion.

- Create sample test uploads (1 audio, 1 video) (1 hr)

- Run end-to-end tests for upload-to-transcription flow (2 hrs)
- Verify transcript content and format quality (1 hr)
- Write unit tests for upload endpoint and transcription logic (3 hrs)

Total for User Story 3: 7 hours

Team Roles

- Andre: Backend Developer, Transcription Lead
- Aditya: Backend Developer, Cloud Integration
- Paolo: Frontend Developer
- Pranav: Research & Evaluation, STT/Diarization
- Parul: QA & Testing

Initial Task Assignment

- Andre: User Story 2 – Integrate transcription engine and export transcript output
- Aditya: User Story 1 – Set up backend upload API and cloud storage connection
- Paolo: User Story 1 – Build frontend file upload UI
- Parul: User Story 3 – Create test files and run end-to-end pipeline tests