

Project Specification for Group #9

Team Name:

VisionText

Domain:

Image Text Extraction (OCR) with User Management and History Tracking

Software Specification:

Java Swing application that allows users to create an account and log in. Once logged in, users can upload image files (e.g., photos or screenshots), extract text content from the images using the OCR.space API, view and edit the text, and save it as a .txt file. Additionally, each user has a personal history of all converted texts that can be viewed and downloaded later.

User Stories

User Story #	Description	Assigned Member	Team Story?
1	As a new user, I want to create an account so that I can have a personal profile and store my text conversion history.	Krisvir Aujla	Yes
2	As a returning user, I want to log into my account so that I can access my previous text conversions.	Jok	No
3	As a logged-in user, I want to upload an image file so that I can extract text from it.	Harry Wu	No
4	As a logged-in user, I	Habib	No

	want to view, edit, and save the extracted text as a .txt file so that I can keep or share it.		
5	As a logged-in user, I want to view my past conversion history so I can download or reuse previous outputs.	Team	No
6	As a user, I want to see error messages when login fails or OCR fails so that I can understand what went wrong and fix it.	Team	No

Proposed Entities for the Domain

- User
 - username: String
 - password: String
 - history: List<TextRecord>
- TextRecord
 - id: int
 - textContent: String
 - imageFileName: String
 - timestamp: Date
- ImageFile
 - filePath: String
 - imageData: BufferedImage
- OCRService
 - apiKey: String
 - sendImageForOCR(ImageFile): String
- TextEditor
 - textContent: String
 - editText(newContent: String): void
- FileExporter
 - saveAsTxt(textContent: String, destinationPath: String): void

- ErrorHandler
 - errorMessage: String
 - displayError(message: String): void
- UserManager
 - registerUser(username, password): void
 - loginUser(username, password): User
 - getUserHistory(User): List<TextRecord>
- UserInterface
 - displayLoginScreen(): void
 - displayMainApp(User): void
 - displayImage(ImageFile): void
 - displayText(String): void
 - showError(String): void

Proposed APIs for the project

1. OCR.space API

Link: <https://ocr.space/ocrapi>

Description: Cloud-based OCR service that accepts images and returns extracted text in JSON format. Supports multiple languages and advanced features such as auto-rotation and table recognition.

Advantages:

- No local installation needed.
- Easy integration via HTTP requests.

Disadvantages:

- Requires internet connection.
- Free tier limitations and possible watermarks.

Implementation Notes:

We can use Java's HttpURLConnection or libraries like OkHttp to send images to OCR.space and parse JSON responses with org.json or Gson.

2. Firebase Authentication & Firestore API

Link: <https://firebase.google.com/docs>

Description: Firebase Authentication provides secure sign-up and login functionalities, including password-based login. Firestore is a cloud-hosted NoSQL database to store user history (converted text records).

Advantages:

- Easy to integrate with Java applications via REST API or community Java SDKs.
- Secure authentication with built-in user management.
- Real-time database updates, flexible data model.

Disadvantages:

- Requires internet connection.
- Free tier has usage limits; may incur cost at scale.

Implementation Notes:

Use HTTP requests or Java SDK to manage user registration, login, and storing/retrieving text conversion history.

Scheduled Meeting Times + Mode of Communication

Weekly tutorial meeting: [Insert your lab/tutorial day and time here]

Additional weekly meeting: [Specify day & time — e.g., Thursday at 5 pm]

Mode of communication: Discord or Microsoft Teams (depending on group preference)

Optional: Team Roles

- Project Manager (Krisvir Aujla): Oversees project planning, task assignments, and progress tracking.
- Backend & API Integration (Jok): Implements OCR API integration and user authentication/database logic with Firebase.
- Frontend & UI (Harry Wu): Designs and implements the Java Swing interface including login screens, main dashboard, and history views.
- Database & History Tracking (Habib): Implements user history storage in Firestore and retrieval functionalities.