CHP 4:

ARCHITECTURE DIAGRAM:

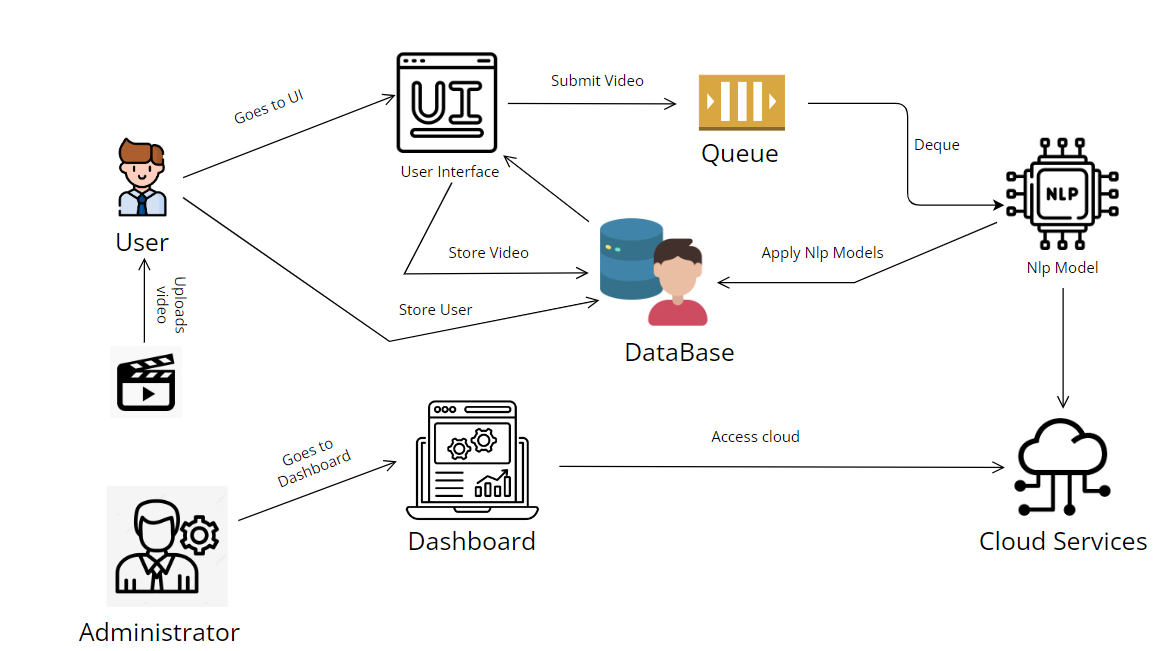
This architecture diagram outlines the components of a Sign Language Transcription System:

1. **User**: Represents individuals interacting with the system, providing user data and submitting sign language videos.
2. **SignLanguageVideo**: Represents videos of sign language submitted by users.
3. **UserDatabase**: Stores user information and sign language videos.
4. **UserInterface**: Web interface for users to interact with the system, displaying user information, and submitting sign language videos.
5. **MessageQueue**: Manages the queue of sign language videos for processing.
6. **TranslationService**: Utilizes NLP models to process sign language videos and generate text.
7. **CloudServices**: Integrates the message queue and translation service.
8. **Administrator**: Represents administrators with access to system management.
9. **AdministratorInterface**: Web interface for administrators to access the dashboard, select actions, and apply changes.

Key Interactions:

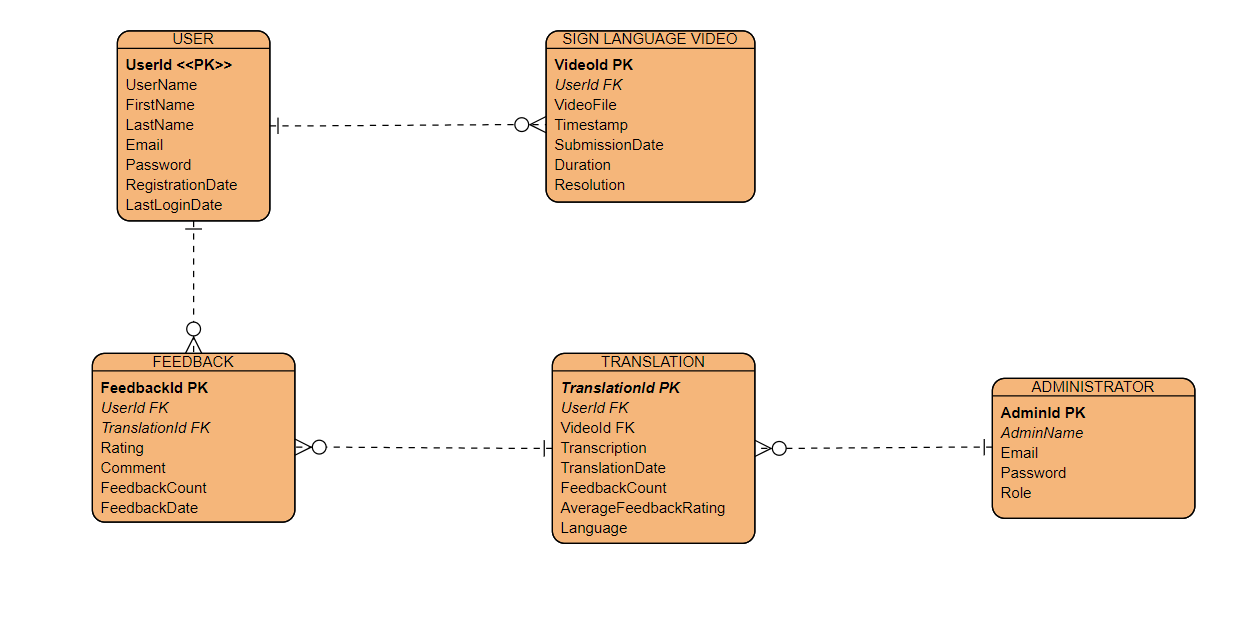
* Users submit sign language videos through the UserInterface.
* User data and videos are stored in the UserDatabase.
* MessageQueue manages the queue of videos for processing.
* TranslationService processes videos using NLP models.
* CloudServices integrate the message queue and translation service.
* Administrators manage the system through the AdministratorInterface.

This architecture facilitates the transcription of sign language videos into text, promoting inclusivity and accessibility.



ERD:

1. **User Entity:**
   * **Attributes:** **UserId**, **UserName**, **FirstName**, **LastName**, **Email**, **Password**, **RegistrationDate**, **LastLoginDate**, and other user-specific attributes.
   * **Relationships:**
     + A User can submit multiple SignLanguageVideos ("Submits").
     + A User can provide feedback on multiple Translations ("Provides Feedback").
2. **SignLanguageVideo Entity:**
   * **Attributes:** **VideoId**, **UserId** (foreign key), **VideoFile**, **Timestamp**, **SubmissionDate**, **Duration**, **Resolution**, and other video-specific attributes.
   * **Relationships:**
     + A User can submit SignLanguageVideos ("Submits" relationship).
     + Each SignLanguageVideo is associated with a Translation through **VideoId**.
3. **Administrator Entity:**
   * **Attributes:** **AdminId**, **AdminName**, **Email**, **Password**, **Role**, and other admin-specific attributes.
   * **Relationships:**
     + An Administrator can review multiple Translations ("Reviews" relationship).
4. **Translation Entity:**
   * **Attributes:** **TranslationId**, **VideoId** (foreign key), **Transcription**, **TranslationDate**, **FeedbackCount**, **AverageFeedbackRating**, **Language**, and other translation-specific attributes.
   * **Relationships:**
     + A Translation is associated with a SignLanguageVideo through **VideoId**.
     + A Translation can have multiple Feedback entries ("Has" relationship).
5. **Feedback Entity:**
   * **Attributes:** **FeedbackId**, **UserId** (foreign key), **TranslationId** (foreign key), **Rating**, **Comment**, **FeedbackDate**, and other feedback-specific attributes.
   * **Relationships:**
     + A User can provide feedback on Translations ("Provides Feedback" relationship).
     + A Translation can have multiple Feedback entries ("Has" relationship).



DATA DICTIONARY:

User Entity

|  |  |  |
| --- | --- | --- |
| FEILD | Data Type | Description |
| UserId (PK) | INT | Unique identifier for a user. |
| UserName | VARCHAR | User's username. |
| FirstName | VARCHAR | User's first name. |
| LastName | VARCHAR | User's last name. |
| Email | VARCHAR | User's email address. |
| Password | VARCHAR | User's hashed password. |
| RegistrationDate | DATE | Date when the user registered in the system. |
| LastLoginDate | DATE | Date of the user's last login. |

SignLanguageVideo Entity

|  |  |  |
| --- | --- | --- |
| FEILD | Data Type | Description |
| VideoId (PK) | INT | Unique identifier for a sign language video. |
| UserId (FK) | INT | Foreign key referencing the User entity. |
| VideoFile | VARCHAR | File path or identifier for the sign language video. |
| Timestamp | DATETIME | Timestamp of when the video was submitted. |
| SubmissionDate | DATE | Date when the video was submitted. |
| Duration | INT | Duration of the sign language video in seconds. |
| Resolution | VARCHAR | Resolution of the video (e.g., HD, 4K). |

ADMINISTRATOR ENTITY:

|  |  |  |
| --- | --- | --- |
| FEILD | Data Type | Description |
| AdminId (PK) | INT | Unique identifier for an administrator. |
| AdminName | VARCHAR | Administrator's name. |
| Email | VARCHAR | Administrator's email address. |
| Password | VARCHAR | Administrator's hashed password. |
| Role | VARCHAR | Role or position of the administrator. |

Translation Entity:

|  |  |  |
| --- | --- | --- |
| FEILD | Data Type | Description |
| TranslationId (PK) | INT | Unique identifier for a translation. |
| VideoId (FK) | INT | Foreign key referencing the SignLanguageVideo entity. |
| Transcription | TEXT | Transcription of the sign language video. |
| TranslationDate | DATE | Date when the translation was generated. |
| FeedbackCount | INT | Number of feedback entries for the translation. |
| AverageFeedbackRating | FLOAT | Average rating from user feedback for the translation. |
| Language | VARCHAR | Language of the transcription/translation (e.g., English). |

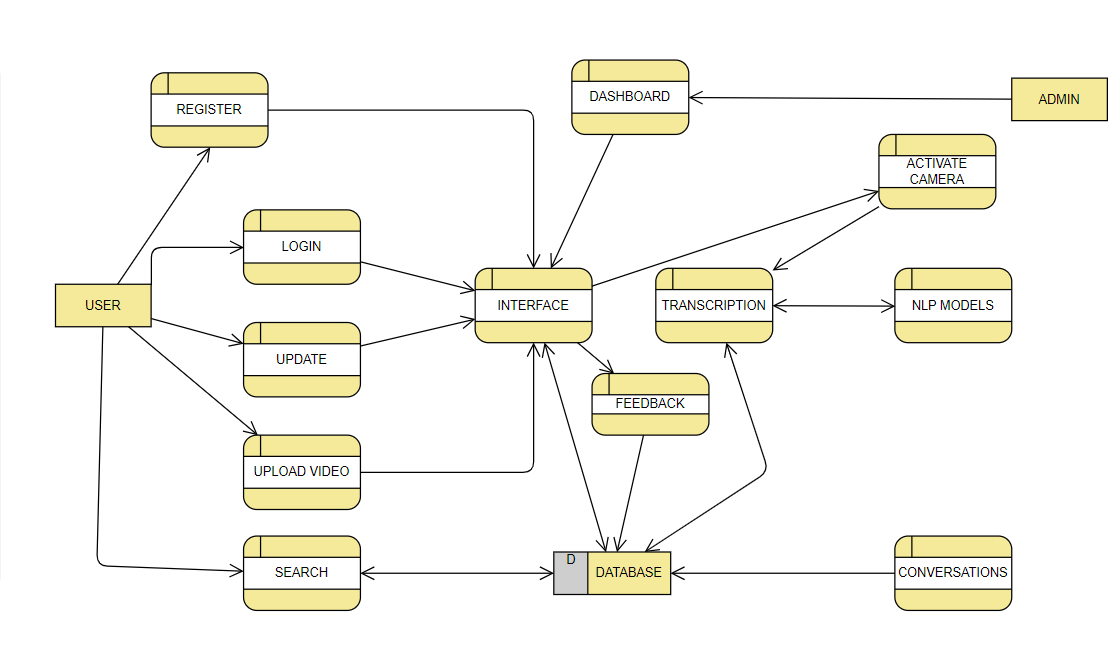
Feedback Entity:

|  |  |  |
| --- | --- | --- |
| FEILD | Data Type | Description |
| FeedbackId (PK) | INT | Unique identifier for a feedback entry. |
| UserId (FK) | INT | Foreign key referencing the User entity. |
| TranslationId (FK) | INT | Foreign key referencing the Translation entity. |
| Rating | INT | User's rating for the translation (e.g., 1 to 5). |
| Comment | TEXT | User's comments or feedback on the translation. |
| FeedbackDate | DATE | Date when the feedback was submitted. |

DFD:

1. **User (External Agent):**
   * Represents the external user interacting with the Sign Language Transcription System.
2. **User Interface Process:**
   * **Responsibilities:**
     + Handles user interactions and requests.
   * **Functions:**
     + Accepts user input for video submission, feedback, and other system interactions.
     + Provides a user-friendly interface for seamless interaction.
   * **Output:** Passes user requests and data to the Sign Language Video Processing and Feedback Handling System.
3. **Sign Language Video Processing Process:**
   * **Responsibilities:**
     + Processes sign language videos submitted by users.
   * **Functions:**
     + Extracts key features from sign language videos, including hand movements, facial expressions, and body language.
     + Prepares the video data for further analysis and translation.
   * **Output:** Sends processed video data to the Translation and Transcription Engine.
4. **Translation and Transcription Engine Process:**
   * **Responsibilities:**
     + Utilizes advanced Natural Language Processing (NLP) models to transcribe sign language into natural language text.
   * **Functions:**
     + Applies NLP techniques to understand sign language gestures and expressions.
     + Translates sign language features into coherent and understandable natural language text.
   * **Output:** Provides the transcribed text for further use and analysis.
5. **Feedback Handling System Process:**
   * **Responsibilities:**
     + Manages user feedback for system improvement.
   * **Functions:**
     + Collects and processes user feedback on transcriptions and system performance.
     + Analyzes feedback to identify areas for improvement.
   * **Output:** Implements system enhancements based on user feedback.

In summary, this DFD outlines the main components and their interactions in the Sign Language Transcription System. Users interact with the User Interface to submit videos and provide feedback. The system processes videos, applies advanced NLP for transcription, and manages feedback to enhance system performance. The directional arrows represent the flow of data and control between these components, providing a visual representation of the system's functionality.



CLASS DIAGRAM: