**API Documentation**

1. Login

* **URL**: **/login**
* **Method**: **POST**
* **Auth required**: No
* **Headers**: **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "username": "user", "password": "password" }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**:

jsonCopy code

{ "token": "token\_value", "userId": 1 }

* **Error Response**:
  + **Code**: **400 Bad Request**
  + **Content**:

jsonCopy code

{ "error": "User name or password error!" }

2. Sign Up

* **URL**: **/signup**
* **Method**: **POST**
* **Auth required**: No
* **Headers**: **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "username": "newuser", "email": "email@example.com", "password": "newpassword" }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**:

jsonCopy code

{ "message": "User created successfully" }

* **Error Response**:
  + **Code**: **400 Bad Request**
  + **Content**:

jsonCopy code

{ "message": "Username already exists" }

3. Download YouTube Video

* **URL**: **/download-youtube**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "youtube\_url": "https://youtube.com/video\_url", "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: Detailed information about the download, subtitles, and processing status.
* **Error Response**:
  + **Code**: **400 Bad Request**
  + **Content**: Specific error message based on failure reason.

4. Speech Recognition

* **URL**: **/speech-recognition**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: multipart/form-data**
* **FormData**:
  + **audio**: The audio file.
  + **request\_type**: "update" or "insert".
  + **sub\_subtitle\_id**: Subtitle ID if updating.
  + **id**: Record ID if updating.
  + **user\_id**: User's ID.
* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: Transcription details.
* **Error Response**:
  + **Code**: **400 Bad Request**
  + **Content**: Error message.

5. ChatGPT Interaction

* **URL**: **/chatGPT**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "request\_type": "update", "id": 1, "sub\_subtitle\_id": 1, "mission\_type": "type\_here", "prompt": "Your prompt here", "model": "gpt-4", "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: ChatGPT's response.
* **Error Response**: Depends on the specific error encountered.

6. Insert Sub-Subtitle

* **URL**: **/insert-sub-subtitle**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "youtube\_id": "youtube\_video\_id", "start\_time": "start\_time\_in\_seconds", "end\_time": "end\_time\_in\_seconds", "text": "Subtitle text", "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**:

jsonCopy code

{ "id": 1, "startTime": "start\_time\_in\_seconds", "endTime": "end\_time\_in\_seconds", "text": "Subtitle text" }

* **Error Response**: Specific error message based on failure reason.

7. Delete Sub-Subtitle

* **URL**: **/delete-sub-subtitle**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "sub\_subtitle\_id": 1, "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**:

jsonCopy code

{ "message": "SubSubtitle updated successfully" }

* **Error Response**:
  + **Code**: **404 Not Found**
  + **Content**:

jsonCopy code

{ "error": "SubSubtitle with this id does not exist" }

8. Remove Bookmark

* **URL**: **/remove\_bookmark**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "youtube\_id": "youtube\_video\_id", "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**:

jsonCopy code

{ "success": "Bookmarks with this id have been deleted" }

* **Error Response**:
  + **Code**: **404 Not Found**
  + **Content**:

jsonCopy code

{ "error": "Bookmark with this id does not exist" }

9. Add Bookmark

* **URL**: **/add\_bookmark**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "youtube\_id": "youtube\_video\_id", "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**:

jsonCopy code

{ "success": "Bookmark with id youtube\_video\_id has been created" }

* **Error Response**:
  + **Code**: **500 Internal Server Error**
  + **Content**: Specific error message based on failure reason.

10. Select All Bookmarks

* **URL**: **/select\_all\_bookmarks**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: Array of bookmark objects with **youtube\_id**, **youtube\_name**, and **youtube\_duration**.
* **Error Response**:
  + **Code**: **404 Not Found**
  + **Content**: Specific error message based on failure reason.

11. Get Statistic

* **URL**: **/get\_statistic**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: Detailed statistics including dates, categories, video names, durations, and extra experience points.
* **Error Response**: Specific error message based on failure reason.

12. Get Tasks

* **URL**: **/get\_tasks**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: Detailed tasks information including daily and weekly tasks with their progress and completion status.
* **Error Response**: Specific error message based on failure reason.

13. Shut Down

* **URL**: **/shut\_down**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**: None
* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: None (The action is to shut down the server, so a successful response might not be practically receivable.)
* **Error Response**: N/A (This endpoint is expected to shut down the system and might not provide a response.)

14. Import API Key

* **URL**: **/import\_api\_key**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "user\_id": 1 }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: The API key.
* **Error Response**: Specific error message based on failure reason.

15. Edit API Key

* **URL**: **/edit\_api\_key**
* **Method**: **POST**
* **Auth required**: Yes (Token Authentication)
* **Headers**: **Authorization: Token <your\_token\_here>**, **Content-Type: application/json**
* **Body**:

jsonCopy code

{ "user\_id": 1, "api\_key": "new\_api\_key\_value" }

* **Success Response**:
  + **Code**: **200 OK**
  + **Content**: Success message.
* **Error Response**: Specific error message based on failure reason.