

# GTA Hiring and Management System - User Manual

## 1. Overview

The GTA Management System is a web-based application designed to streamline the management of Graduate Teaching Assistant (GTA) assignments. It simplifies workflows with features like course data import, dynamic GTA request generation, approval sheet management, cross-listing detection, and an **editable approval sheet interface** for seamless modification and approval.

## 2. Prerequisites

Before running the application, ensure you have the following installed:

- **Python 3.x**
- **pip** (Python package installer)
- **MongoDB** (for database operations)
- A compatible browser (e.g., Chrome, Firefox)

## 3. Installation Steps

### 1. Clone the Repository

```
bash
```

Copy code

```
git clone <repository_url>
```

```
cd Milestone_3_finalcode
```

### 2. Install Dependencies

Install all required Python libraries using the requirements.txt file:

```
bash
```

Copy code

```
pip install -r requirements.txt
```

### 3. Setup the Database

➤ Ensure MongoDB is running locally or remotely.

➤ Import the initial catalog data by running:

```
bash
```

Copy code

```
python insertcatalogcourse.py
```

➤ Confirm that the data has been added to the database.

### 4. Run the Application

Start the Flask application by running:

```
bash
```

Copy code

```
python app.py
```

The application will be available at <http://127.0.0.1:5000/>.

## 4. Application Features

### Frontend Pages

- **Login Page:** URL: /login
  - Users authenticate to access the system.
  - Forgot password functionality is available.
- **Create Account Page:**  
URL: /create\_account
  - Allow new users to create accounts.
- **Import Course Data Page:**  
URL: /import\_course\_data
  - Upload course data (.xlsx) to the system.
- **Generate GTA Request Page:**  
URL: /generate\_gta\_request
  - Automatically generates GTA assignment requests based on uploaded course data and predefined rules (gta\_rules.json).
- **View Approval Sheets:**  
URL: /view\_approval\_sheet
  - Displays all approval sheets for review.
- **Upload Approval Sheet:**  
URL: /upload\_approval\_sheet
  - Allows users to upload finalized approval sheets.
- **Editable Approval Sheet:**  
URL: /editable\_approval\_sheet
  - Provides an interface to edit generated approval sheets in real-time.
  - **Features of Editable Approval Sheet:**
    - Modify GTA assignments and related details directly in the browser.
    - Save changes dynamically to the database.
    - Export the updated sheet for documentation or sharing.

## 5. Key Files and Folders

- **Templates Folder:** Contains all HTML templates for the web pages.
  - editable\_approval\_sheet.html: UI for editing approval sheets.
  - Other templates: Login, upload, and data management pages.
- **app.py:** Core backend logic and route definitions.
- **catalog\_course.xlsx:** Initial course data to import into the system.

- **gta\_rules.json**: JSON file defining rules for GTA requests.
- **insertcatalogcourse.py**: Script for loading course data into MongoDB.
- **uploads Folder**: Stores uploaded files, like approval sheets.

## 6. User Workflow

- **Log in or Create an Account:**
  - Navigate to /login to access the system with your credentials.
  - If you're a new user, use /create\_account to register.
- **Import Course Data:**
  - Navigate to /import\_course\_data.
  - Upload the catalog\_course.xlsx file containing the course and section data.
  - The system will validate and store the data in the database.
- **Generate GTA Requests:**
  - Navigate to /generate\_gta\_request.
  - The system automatically generates GTA assignment requests based on uploaded course data and predefined rules (gta\_rules.json).
  - Preview of the generated requests to confirm accuracy.
- **Upload Approval Sheet (Hiring Management Module):**
  - Navigate to /upload\_approval\_sheet.
  - Upload the finalized or system-generated approval sheet to integrate the hiring management module with detailed GTA data.
- **Edit Approval Sheet:**
  - After uploading, proceed to /editable\_approval\_sheet to finalize GTA assignments.
  - Features include:
    - **Add/Select Student Assignments**: Assign or reassign GTA slots to students, including all relevant student details (e.g., name, ID, email, and Phone number).
    - **Edit Student Details**: Modify student-specific data such as contact information, course assignments, or assigned hours.
    - **Real-Time Updates**: Ensure changes are reflected immediately in the database.
  - Once finalized, save the approval sheet for further processing.
- **Export Data:**
  - Export generated requests, approval sheets, or other data as .xlsx or .csv files for reporting or sharing.
- **Log Out:**
  - After completing tasks, log out securely to protect your account.

## 7. Error Handling

- **File Upload Issues:** Ensure files are in the correct format (.xlsx for course data). The system validates file contents upon upload.
- **Database Errors:** Verify that MongoDB is running and accessible.
- **Login Failures:** Use the "Forgot Password" feature to reset credentials.

### GTA Request Generation Issues:

- Ensure the gta\_rules.json file is correctly formatted and stored in the correct directory.
- Validate the file content with a JSON formatter tool to confirm proper syntax.
- Provide a fallback message if the file is not found or unreadable:
  - Example: *"GTA rules file not found or improperly formatted. Please verify the file."*

## 8. Maintenance

- **Update Dependencies:**

Periodically update libraries using:

```
bash
```

```
Copy code
```

```
pip install --upgrade -r requirements.txt
```

- **Modify Rules:**

Update gta\_rules.json to reflect new assignment rules or multipliers.

- **Backup Database:**

Regularly back up MongoDB to avoid data loss:

```
bash
```

```
Copy code
```

```
mongodump --out <backup_directory>
```