



Technical Documentation SDE Intern (Platform)

PREPARED FOR

Summer Hacker Camp 2019

Innovaccer

PREPARED BY

Sahil Chilana

sahilchilana@gmail.com

+91-8968684862

1. Project Overview

The script supports the following features:

- Accepts a list of user email ids and their show preferences.
- The details input by the user are stored in a SQLite DB.
- The show details are fetched using APIs and scraping the IMDB page for the show.
- The script can be scheduled to send reminders via email for the same set of users.
- The email template is customisable.
- The time at which the emails will be sent out is customisable.

2. Setup

The script can be set up using the following steps:

- Download or clone the repository from <_repo_link_>
- Navigate to the directory where the script is installed.
- Set up a virtual environment using virtualenv <env_name> and activate it using source <env_name>/bin/activate (optional)
- Install dependencies using “pip install -r requirements.txt”

3. User Guide

- The script has the following configuration parameters (line 6 in main.py):
 - DB_PATH : The relative or absolute path to the SQLite DB.
 - EMAIL_TEMPLATE: The template in which the reminder is to be sent. The placeholders for the show name and status are _show_name_ and _status_ respectively.
 - EMAIL_TIME: The time at which emails will be sent out in 24 hours time format. Eg: “10:30”
 - The script may prompt for credentials when running for the first time.
 - Username: hypernova0610@gmail.com
 - Password: Playhard76

4. Architecture

- The script has a modular architecture where in each module is independent in functionality with respect to the other modules.
- The script comprises of 3 modules:

- `fetch_details.py`
 - The module interfaces using the function using `getShowStatus(show_name)`
 - The IMDb id of the title is fetched using the OMDb APIs.
 - The IMDb page of the title and the seasons page is scraped to fetch the status.
- `send_email.py`
 - The module makes use of the Gmail API to send out email reminders to users from the email hypernova0610@gmail.com
 - It interfaces with the main module using the function `sendEmailReminder(reciver_email, email_body)`
- `main.py`
 - The module is responsible for coordinating with the other modules and updating the DB.
 - The script prompts the user to enter 1 to add/update users to the DB or run in scheduled mode.
 - The scripts assumes that valid email ids will be entered and valid show names will be entered.
 - The script updates the user details. Duplicates are ignored automatically.

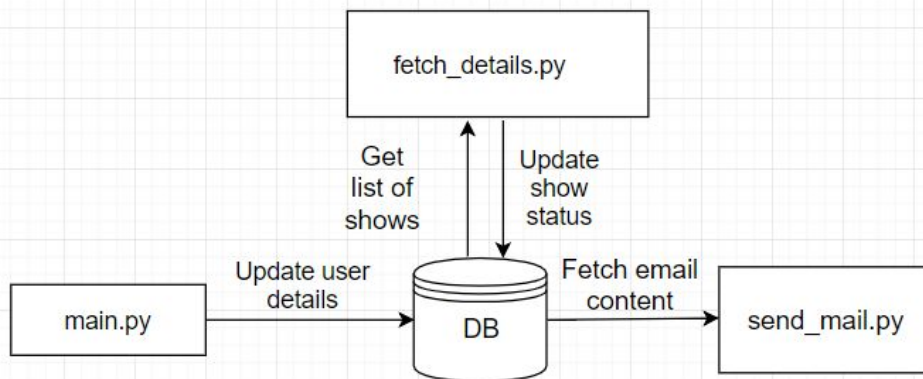


fig : Script Architecture

- The script uses a DB store to optimise the performance and reduce the number of redundant requests sent to fetch show status.

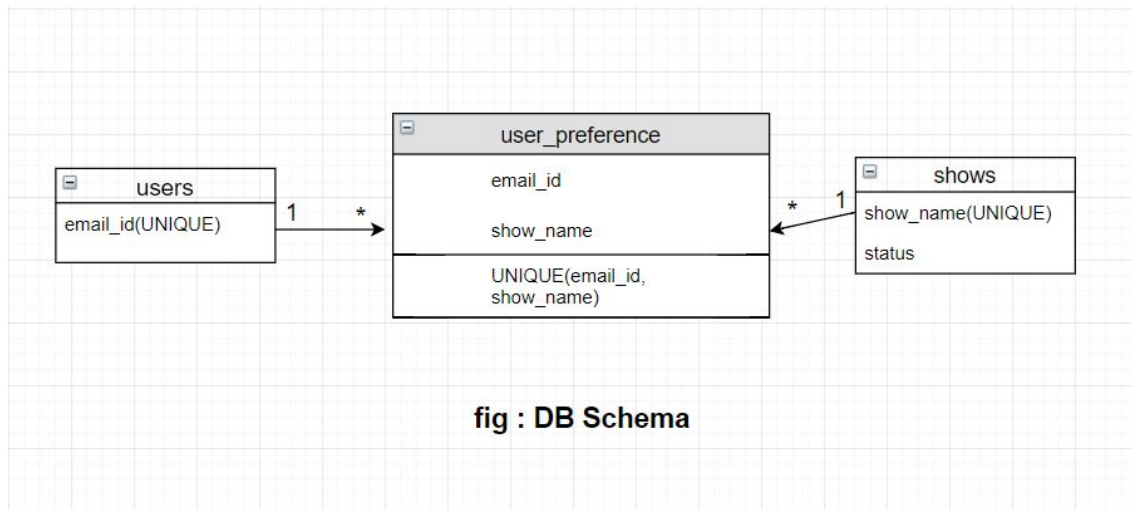


fig : DB Schema

- The unique constraints on the DB ensure that show status needs to be fetched only once.

5. Further Improvements

The following improvements can be made to the script in the future:

- Validating the format of email addresses entered.
- Support for show names containing accented characters.
- Add a calendar object (.ical format) to provide the functionality of directly adding the show as a reminder in the user's calendar.
- Not updating the status in the DB for shows which have been discontinued.

6. Compatibility

The script has been tested to work on UNIX based systems (Mac OS and Linux). The script has not been tested on Windows.