

# Abhay Chaudhary

21F1002435

[21f1002435@student.onlinedegree.iitm.ac.in](mailto:21f1002435@student.onlinedegree.iitm.ac.in)

**About Me:** Experienced Documentation Manager with a demonstrated history of working in the education management industry. Skilled in AutoCAD, Management, Adobe Photoshop, Ethical Hacking, and Research and Development (R&D). Reliable administrative professional pursuing Bachelor of Technology - BTech focused in Computer Science from Vellore Institute of Technology.

## Description

Technologies used

- Flask for application code
- Jinja2 templates + Bootstrap for HTML generation and styling
- SQLite for data storage

The following python packages are needed to run this app:

- Flask 2.0.3
- Flask-SQLAlchemy 2.5.1
- Flask-Login 0.5.0
- Bcrypt-Flask 1.0.2
- matplotlib 3.5.1
- Used for self tracking - tracking habits, activities, other life parameters etc.
- User can have multiple trackers
- Each tracker will have a
  - ID
  - Name
  - Description
  - Tracker type
  - Settings
- User can log to one more tracker at any time, each time it's logged it will capture
  - TimeStamp
  - Tracker
  - Value (based on the corresponding tracker type)
  - Note
- System will track progress over time and shows graphs trend lines etc

## DB Schema Design

```
CREATE TABLE "sqlb_temp_table_2" {
  "user_id" INTEGER,
  "username" TEXT NOT NULL UNIQUE,
  "password" TEXT NOT NULL,
  PRIMARY KEY("user_id" AUTOINCREMENT)
)
CREATE TABLE "Selectable_Values" (
  "value_id" INTEGER,
  "tracker_id" INTEGER,
  "selectables" TEXT NOT NULL,
  PRIMARY KEY("value_id" AUTOINCREMENT),
  FOREIGN KEY("tracker_id") REFERENCES "Trackers"("tracker_id")
)
CREATE TABLE "Logs" (
  "log_id" INTEGER,
  "user_id" INTEGER,
  "tracker_id" INTEGER,
  "timestamp" TEXT NOT NULL,
  "value" TEXT NOT NULL,
  "note" TEXT,
  PRIMARY KEY("log_id" AUTOINCREMENT),
  FOREIGN KEY("tracker_id") REFERENCES "Trackers"("tracker_id"),
  FOREIGN KEY("user_id") REFERENCES "Users"("user_id")
)
```

```

)
CREATE TABLE "Enrollments" (
  "enroll_id" INTEGER,
  "user_id" INTEGER,
  "tracker_id" INTEGER,
  FOREIGN KEY("user_id") REFERENCES "Users"("user_id"),
  FOREIGN KEY("tracker_id") REFERENCES "Trackers"("tracker_id"),
  PRIMARY KEY("enroll_id" AUTOINCREMENT)
)
CREATE TABLE "Trackers" (
  "tracker_id" INTEGER,
  "tracker_name" TEXT NOT NULL UNIQUE,
  "tracker_type" TEXT NOT NULL,
  "tracker_desc" TEXT,
  PRIMARY KEY("tracker_id" AUTOINCREMENT)
)
CREATE TABLE "Users" (
  "user_id" INTEGER,
  "username" TEXT NOT NULL UNIQUE,
  "password" TEXT NOT NULL,
  PRIMARY KEY("user_id" AUTOINCREMENT)
)
CREATE TABLE "Users" (
  "user_id" INTEGER,
  "username" TEXT NOT NULL UNIQUE,
  "password" TEXT,
  PRIMARY KEY("user_id" AUTOINCREMENT)
)

```

## API Design

The OpenAPI Specification (OAS) defines a standard, language-agnostic interface to RESTful APIs which allows both humans and computers to discover and understand the capabilities of the service without access to source code, documentation, or through network traffic inspection. When properly defined, a consumer can understand and interact with the remote service with a minimal amount of implementation logic.

An OpenAPI definition can then be used by documentation generation tools to display the API, code generation tools to generate servers and clients in various programming languages, testing tools, and many other use cases.

## Architecture and Features

The folder app contains all the views i.e. logviews, trackerviews, userviews. The folder static contains the css and js folder having all the bootstrap files. The folder template has the html files from user login to other aspects like tracking and editing trackers. App.yaml has the necessary api design.

### Features and Functionalities

- User login
- Dashboard and Trendlines
- Tracker management
- Tracker log events

## Video