### Author

Sandeep Nalla

21f1006125

[21f1006125@student.onlinedegree.iitm.ac.in](mailto:21f1006125@student.onlinedegree.iitm.ac.in)

About: Software Professional with 8 years of industry experience with the following skill sets

* REST API development using Spring Boot/Spring Reactive (Java) framework
* Automation Frontend (Web & Mobile Apps) using Selenium (Java) & REST API using RestAssured (Java)

### Description

Implement Self tracker web app for tracking habits, activities, and other life parameters

* Creating or Updating or Deleting a tracker
* Logging events under a tracker and updating or deleting logged events
* Displaying Stats of a tracker in form of graph & list

### Technologies used

* Flask==2.0.3 – To build dynamic frontend web application
* Flask-RESTful==0.3.9 – To build REST APIs
* Flask-SQLAlchemy==2.5.1 – To connect to DB
* Flask-Security – To allow authenticated users to login & authorized users to perform action
* Jinja2==3.0.3 – To generate HTML programmatically
* HTML – Markup language
* SQLite – Data base management
* Bootstrap – UI framework
* Plotly

Tools used

* PyCharm (IDE)
* DB browser for SQLite

### DB Schema Design

* User – For storing allowed user details to access the app
  + id (Integer, Autoincrement, Primary key)
  + email (String, Unique)
  + username (String)
  + password (String)
  + active (Boolean)
* Role – For storing role details
  + id (Integer, Autoincrement, Primary key)
  + name (String, Unique)
  + description (String)
* roles\_users – For maintaining relation between users & roles
  + id (Integer, Autoincrement, Primary key)
  + user\_id (Integer, Foreign key)
  + role\_id (Integer, Foreign key)
  + Constraints:
    - Foreign key – user\_id – To link user
    - Foreign key – role\_id – To link role
* Tracker – For storing tracker details
  + id (Integer, Autoincrement, Primary key)
  + name (String, unique, non-nullable)
  + description (String, non-nullable)
  + type (String, non-nullable)
  + settings (String)
* Log – For storing logged events
  + id (Integer, Primary key, Autoincrement)
  + timestamp (String, non-nullable)
  + tracker (Integer, non-nullable)
  + value (String, non-nullable)
  + note (String)
  + Constraint:
    - Foreign key - tracker.id – To link logged events to a tracker

### API Design

CRUD operations for tracker & log

* /api/trackers
* /api/trackers/<int:tracker\_id>
* /api/trackers/<int:tracker\_id>/logs
* /api/trackers/<int:tracker\_id>/logs/<int:log\_id>

Swagger:

<https://app.swaggerhub.com/apis/NallaSandeep/iit.mad1.finalproject/0.3/>

### Architecture and Features

Project structure:

* Root
  + application
    - config
    - controllers
    - models
  + db-directory
    - db sqlite file
  + static
    - css
    - js
    - images
  + templates
    - HTML files
  + app.py (Orchestration)

Features:

* User login
* Tracker management (Create, View, Edit, Delete)
  + Dashboard and Trendlines
* Tracker log events (Create, View, Edit, Delete)
* APIs for CRUD operations

### Video

<https://drive.google.com/file/d/1f9BRFhvC0y1vCaVN5ROKQ7I5jtPO47Ua/view?usp=sharing>