Storage Structure

Group-1 Team-2

Vatsal Soni - 2018201005
Darshan Kansagara - 2018201033
Dhawal Jain - 2018201065

Subsystem and APIS

1. Authentication Service

a. API:- /user_authenticate

   i. Input:-
      1. User’s credential (UserName, Password)

   ii. Output:-
      1. Success / Error message

2. Deployment Service

a. API:- /deployService

   i. Input:-
      1. Modelname
      2. FileName
      3. ModelPath
      4. DeployIp
      5. DeployPort
      6. DeployPassword
      7. DeployUserName

   ii. Output:-
      1. Deploy successfull / Error

3. Scheduling Service

a. API:- /ScheduleService

   i. Input:-
      1. Modelname
      2. StartTime
      3. EndTime
      4. Repeat
      5. Count
      6. Interval
      7. Repeat_Period

   ii. Output:-
      1. Successfully scheduled / unable to invoke

4. Invocation Service

a. API:- /invocationService/start

   i. Input:-
      1. Modelname
      2. InputStream
      3. DeployIp
      4. DeployPort
      5. DeployPassword
      6. DeployUserName

   ii. Output:-
      1. Model Up successfully

b. API:- /invocationService/end

   i. Input:-
      1. Modelname
      2. DeployIp
      3. DeployPort
      4. DeployPassword
      5. DeployUserName

   ii. Output:-
      1. Model Down successfully

5. Notification & Action Service

a. API:- /notification
    - b. Input:-
        - i. Predicted output of model
    - c. Output:-
        - i. Run the action code / Send message

6. Logging Service

a. API:- /loggingService
    - i. Input:-
        1. Log message
    - ii. Output:-
        1. Write into log file

7. Monitoring Service

a. API:- /monitoring/model_status
    - i. Input:-
        1. User Id
    - ii. Output:-
        1. Model Name
        2. Gateway / Server Instance (ip and port)
        3. Status (UP / Down)

b. API:- /monitoring/gateway_status
    - i. Input:-
        1. None
    - ii. Output:-
        1. Gateway Id
        2. Sensor list
        3. Sensor status (UP / Down)
        4. Gateway Status (UP / Down)

8. Inferencing Service

a. API:- /inferenceSerivce

   i. Input:-
      1. Usecase (Test data)
      2. Action file

   ii. Output:-
      1. Inferencing rate

9. Authorization Service

a. API:- /authorizationService

   i. Input:-
      1. User credentials

   ii. Output:-
      1. Valid / Invalid
      2. List of services that user can access

Storage structures

All these **Data Stored in DB** and We used in-memory data structure used for our services namely:

List, Dictionary, JSON, Queue,

Model Information :

1. Model Name
2. Scheduling info
3. Gateway information
4. User ID
5. status

Scheduling information :

1. Model Name
2. Deployment socket
3. Deployment username
4. Deployment Machine Uname & Password
5. Starttime
6. Endtime
7. Repeat interval
8. count

Sensor information

1. Sensor ID
2. Gateway ID
3. Sensor type
4. Sensor status

Gateway Information

1. Gateway ID
2. List of Sensor
3. Gateway status

User credential

1. Username
2. Password

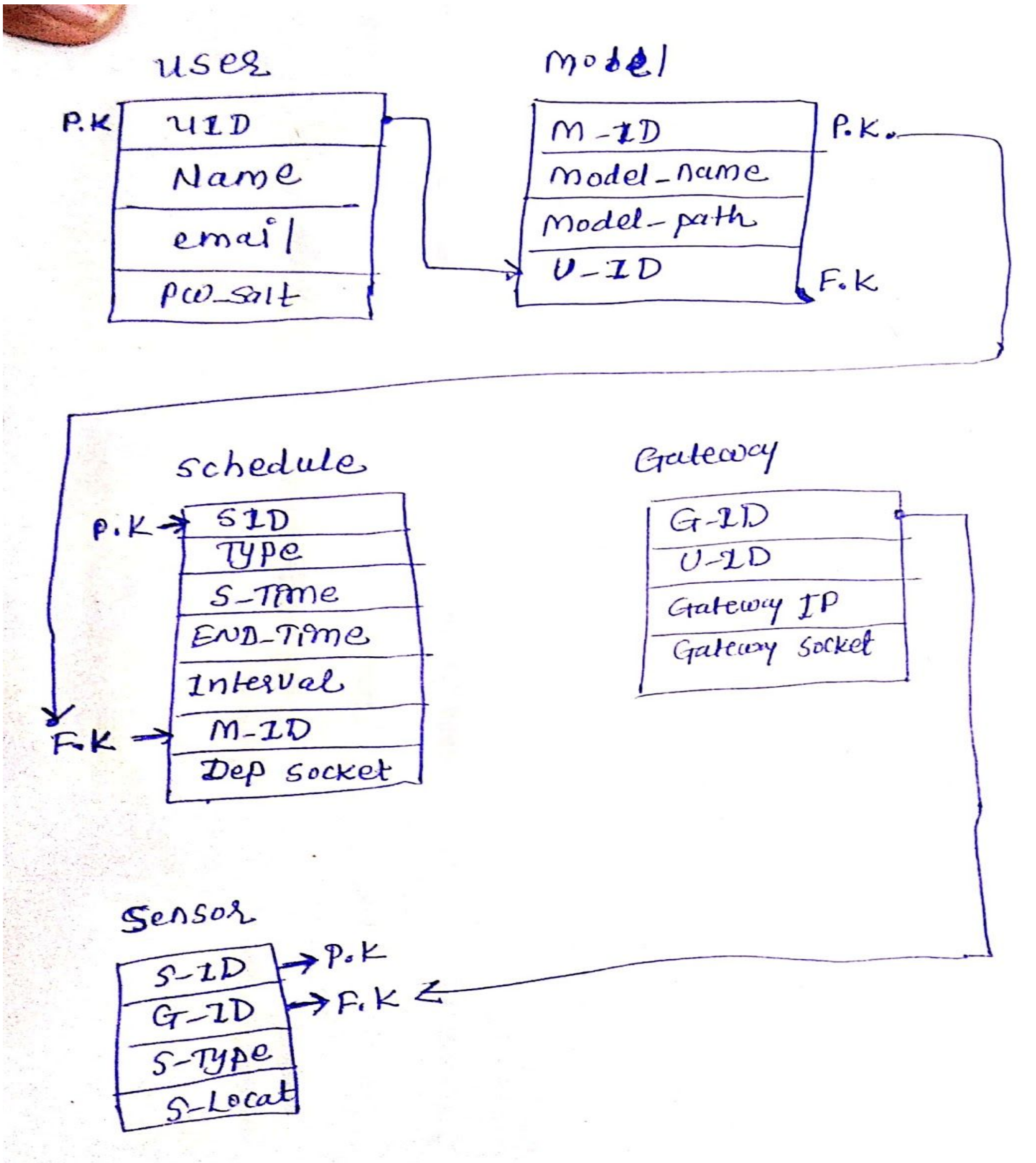
Model data and config file corresponds to a model will be stored in **Network File System(NFS)** that hat lets a computer user view and optionally store and update files on a remote computer as though they were on the user's own computer ad thus allow a very fast access to the shared data on remote machine.

In case of our platform the main 3 files that are widely used by almost every microservice
Are namely:
1 Model files
2 Config files
3 Action files

Format of NFS

In NFS, there is separate folder for each user and each user's model data, config file will be stored in this folder.

DataBase Diagram


user
P.K
UID
Name
email
pw_salt
model
M-ID
model_name
Model-path
U-ID
P.K.
F.K
schedule
P.K
SID
Type
S-Time
END-Time
Interval
M-ID
Dep socket
F.K
Gateway
G-ID
U-ID
Gateway IP
Gateway socket
Sensor
S-ID
G-ID
S-Type
S-Locat
P.K
F.K