SWIPEIO-Requirements Document

SwipeIO

SwipeIO is a web application which tells about the Number of Employees, Average Working Hours and Defaulters which is based on their IN/OUT Timings.

SCREENS:

* Login
* Dashboard
* Report
* Add/Remove
* Settings
* Import

LOGIN**:-**

* Invoke the SwipeIO login page. After invoking login page asks to enter the Email and Password of the Employee.
* The Login can be done by Employees which is categorized as Admins and Contracts.
* The contents of verifying the login page is provided by the database by matching details of Email and Password is correct and authentication is provided by the middleware(i.e. Backend) which tells the employee exists or it should throw error codes based on their invalid data.

**API:**<http://localhost:4200/>Employee/authenticate

Request payload:

{

email:

pass\_word:

}

Response:  
{

status:200

data:{

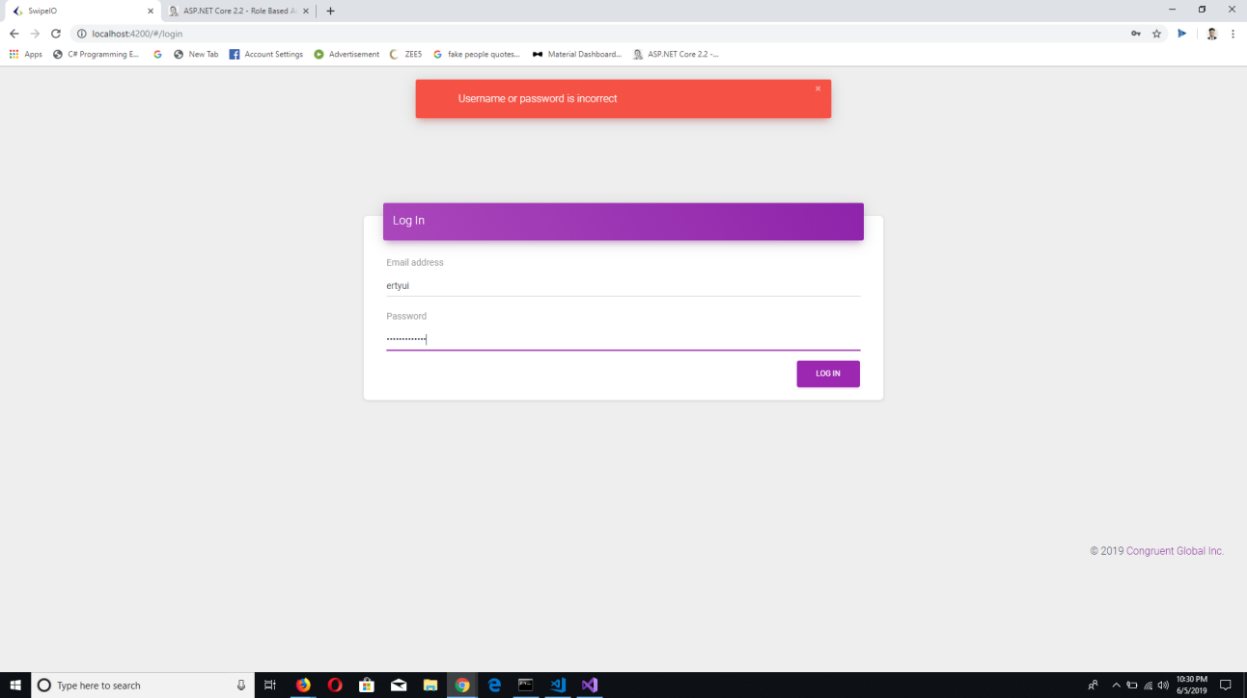
emp\_id,email,emp\_name,emp\_number,is\_admin,card\_id

}

}

**Front end:**

* Use the above api to validate the email id and password provided by the user
* If the user is validated successfully then status will be ‘ok’ with data as employee Id otherwise status as ‘bad’ with error code in the data.

****

**Back end:**

* Implement the API
* Validate the login by passing details to the stored procedure
* Gets the date from the stored procedure and check against the email and password match
* Based on the response of the SP, check if both the email and password match with response of SP ,send the status as ‘ok’ and data as emp\_Id.
* Validate for the email and password and send the respective error code , refer Error code and error messages and send status as ‘bad’

**Database end:**

* Create a stored procedure with the following details

**SP Name** : Validate

**Params**:

email1– string

pass\_word1 – string

**Output**:

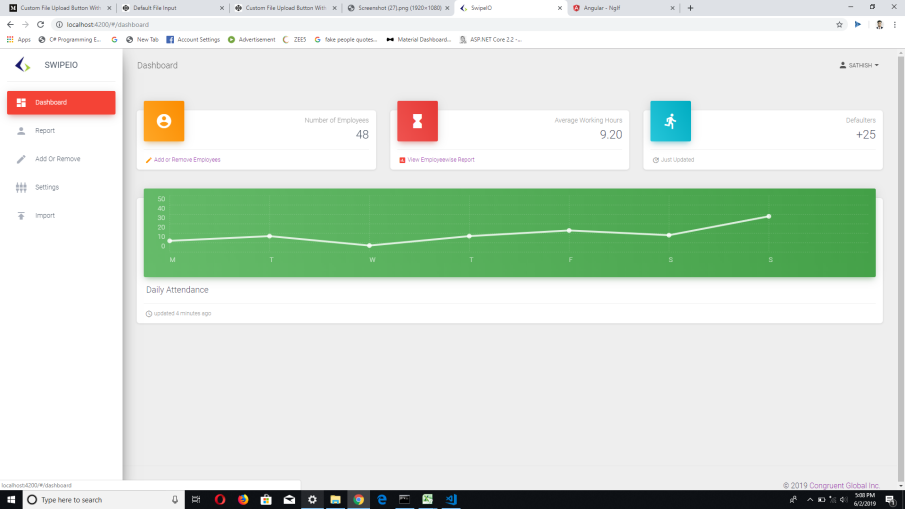
Data of matched rows.

# Error code and error messages

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Sub category** | **Error Code** | **Error Message** |
| Login |  | L01 | Email or password is incorrect |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

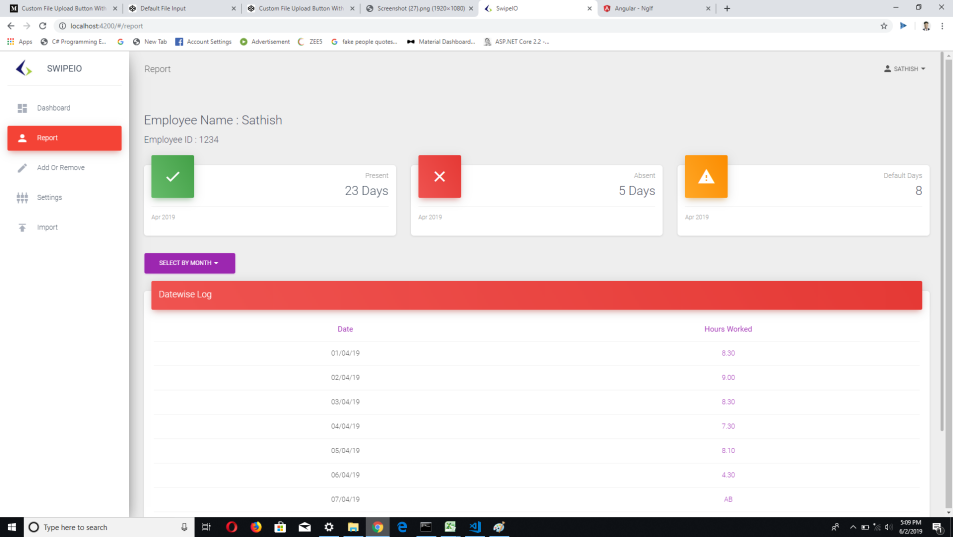
# Dashboard:

* In the Home page of Admin it shows details of all Employees:
* Number of Working Hours
* Average working Hours
* Defaulters
* Graph will be shown overall of how many employees is present on particular day.



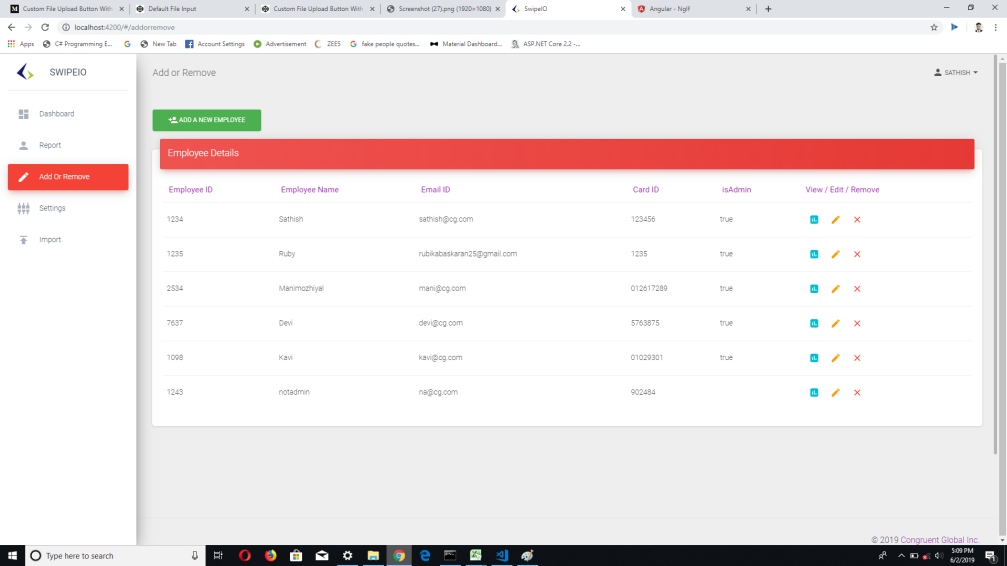
# REPORT:

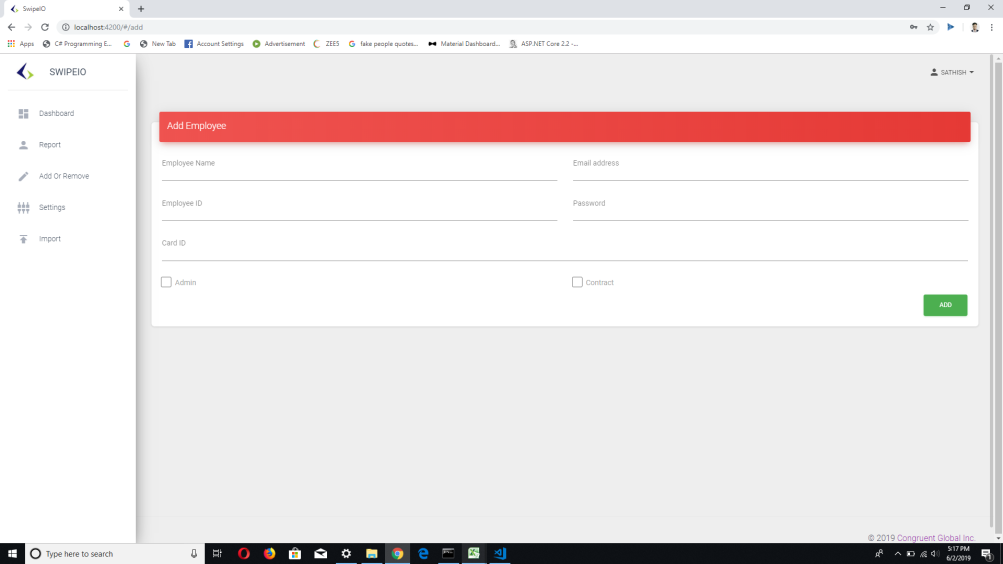
* In the Home Page of Employees & Admin it shows employee:
* Number of Days Present
* Number of Days Absent
* Number of Default Days
* It also shows Date wise Log of employees with date and how many hours worked.



# ADD/REMOVE:

* In Add/Remove the admin can add Employees and can do changes like:
* View
* Edit
* Remove
* To Add a New Employee:
* Employee Name, Employee ID, Email Address, Password, Card Id, Isadmin and contract should be filled.





# Settings:

* In Settings, Gate, Card Details and leave details will be specified of employees.
* And do changes like Edit/Remove of the gate or card details.

# IMPORT:

* In Import screen the admin should upload XLSX File of the employees, based on that they can update the number of hours worked(calculated on IN/OUT) Timings.

**API:**<http://localhost:4200/>Log/upload

Request payload:

{

Date:

Time:

Cardid:

Empid:

EmpName:

Department:

Type:

CID:

Gate:

InOut:

Remark:

}

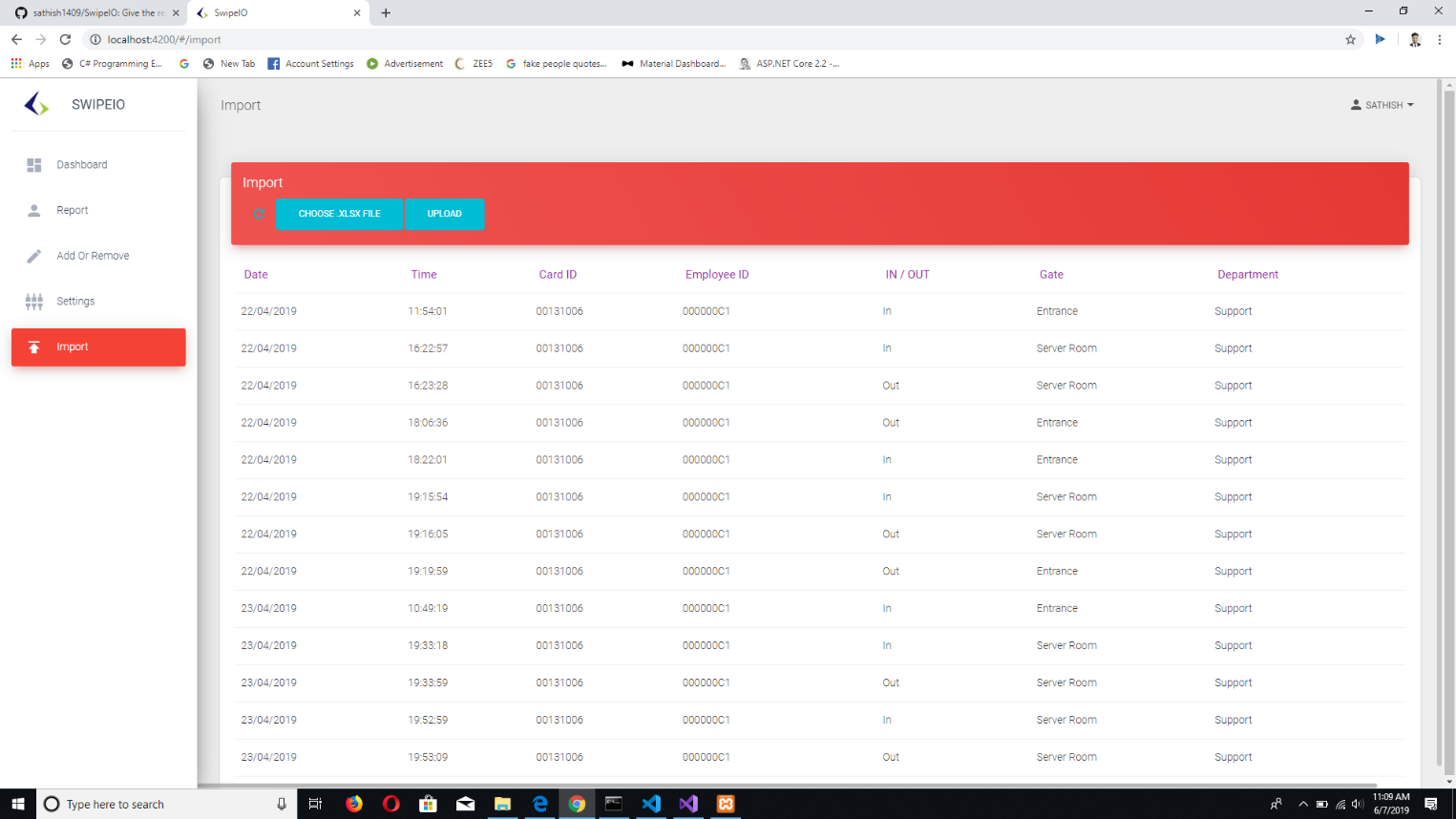
Response:  
{

status:200

}

**Front end:**

* The front-end must have XLSX file when the user clicks that button they must be able to choose the excel file which has log of swipes.
* The front -end should show the contents of the chosen excel file.



**Back end:**

* Implement the API
* Get the log details from front-end and iterate every data and for every iteration call the stored procedure import \_to\_swipe

**Database end:**

* Create a stored procedure with the following details

**SP Name** : import\_to\_swipe

**Params**:

date1, time1, card\_number1, emp\_number1 , gate\_name1 , inorout1 ,in remark1.

**Output**:

Acknowledgement.

# Error code and error messages

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Sub category** | **Error Code** | **Error Message** |
| Import |  | I01 | Failed to import. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |