

**PROJECT SOCCERSCORE**

**วิชา Advanced Web Technology and Protocols รหัสวิชา 060213742**

**จัดทำโดย**

**นายวุฒิพงศ์ จันทร์น้ำ รหัสนักศึกษา 61-060216-1202-5**

**นายกฤษฎีกา วงษ์สุบรรณ์ รหัสนักศึกษา 61-060216-1119-3**

**นายธนธรณ์ เปรมปราณี รหัสนักศึกษา 61-060216-1221-1**

**นางสาวเสาวรจน์ สมเหมาะ รหัสนักศึกษา 61-060216-1207-6**

**นางสาวกรพินธุ์ อินทร์เฟื่อง รหัสนักศึกษา 61-060216-2201-2**

**เสนอ**

**อาจารย์อนิราศ มิ่งขวัญ**

**ภาคเรียนที่ 2/2563**

**คณะเทคโนโลยีและการจัดการอุตสาหกรรม**

**มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ**

**คำนำ**

รายงานเล่มนี้เป็นส่วนหนึ่งของวิชา AdvancedWeb Technology and Protocols รหัสวิชา 060213742 โดยมีจุดประสงค์เพื่อรายงานผลการทำโปรเจค SOCCERSCORE ที่เป็นการดึง API จากภายนอกมาใช้งานในการทำโปรเจค มีการใช้ Mongodb มาเก็บข้อมูลโดยได้มีการนำความรู้ที่ได้เรียนมา มาประยุกต์ใช้ในการทำโปรเจคนี้อีกด้วย

ทางคณะผู้จัดทำหวังเป็นอย่างยิ่งว่า รายงานฉบับนี้จะมีประโยชน์ไม่มากก็น้อยและถ้าหากว่ามีข้อผิดพลาดประการใดทางคณะผู้จัดทำก็ต้องขออภัยมา ณ ที่นี้ด้วย

คณะผู้จัดทำ

26/04/2564

**ขั้นตอนการติดตั้งโปรเจค SOCCERSCORE**

1.ติดตั้ง Python

2.ติดตั้ง Node.js

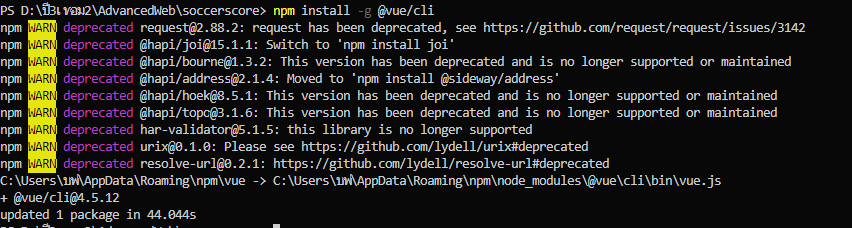
3.ติดตั้ง MongoDB ถ้ามี Database อยู่บน Cloud ก็ไม่ต้องติดตั้ง

4.ทำการ Clone Project จาก GitHub เมื่อ Clone เสร็จแล้ว ให้ทำการใช้คำสั่งดังนี้

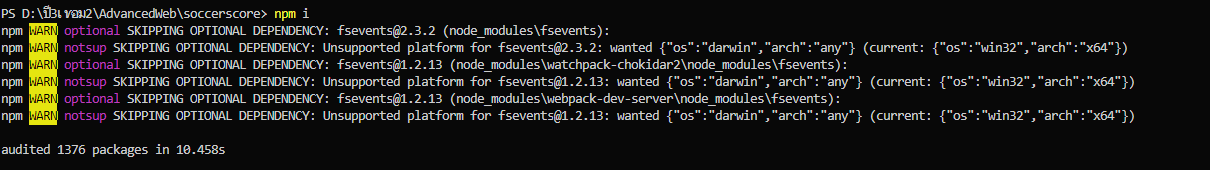
* nmp -v (เอาไว้เช็คว่ามี npm แล้ว)



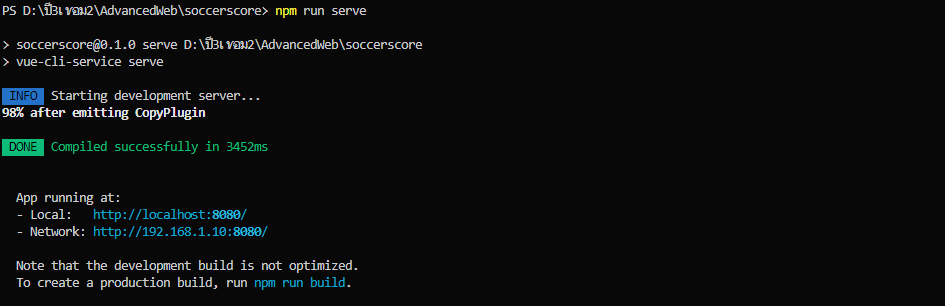
* npm install -g @vue/cli



* npm i

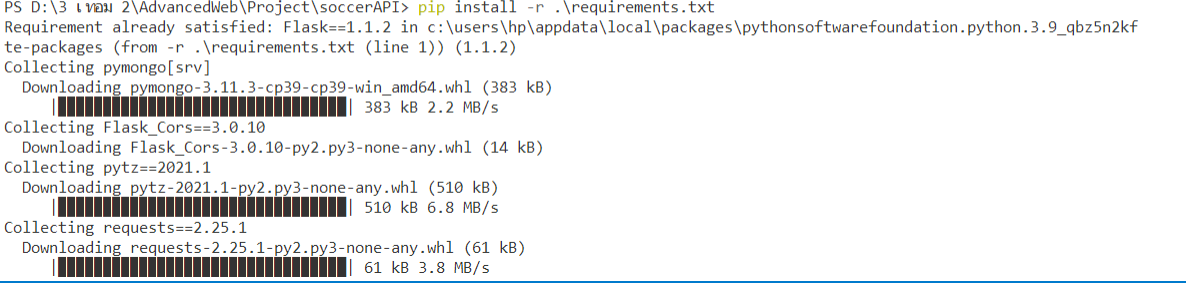


* npm run serve

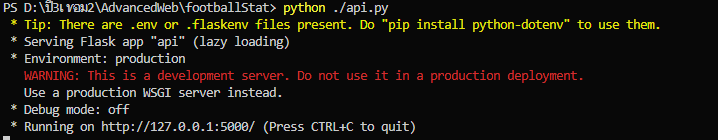


5.ทำการ Clone api เมื่อ Clone เสร็จแล้ว

* pip install -r .\requirements.txt



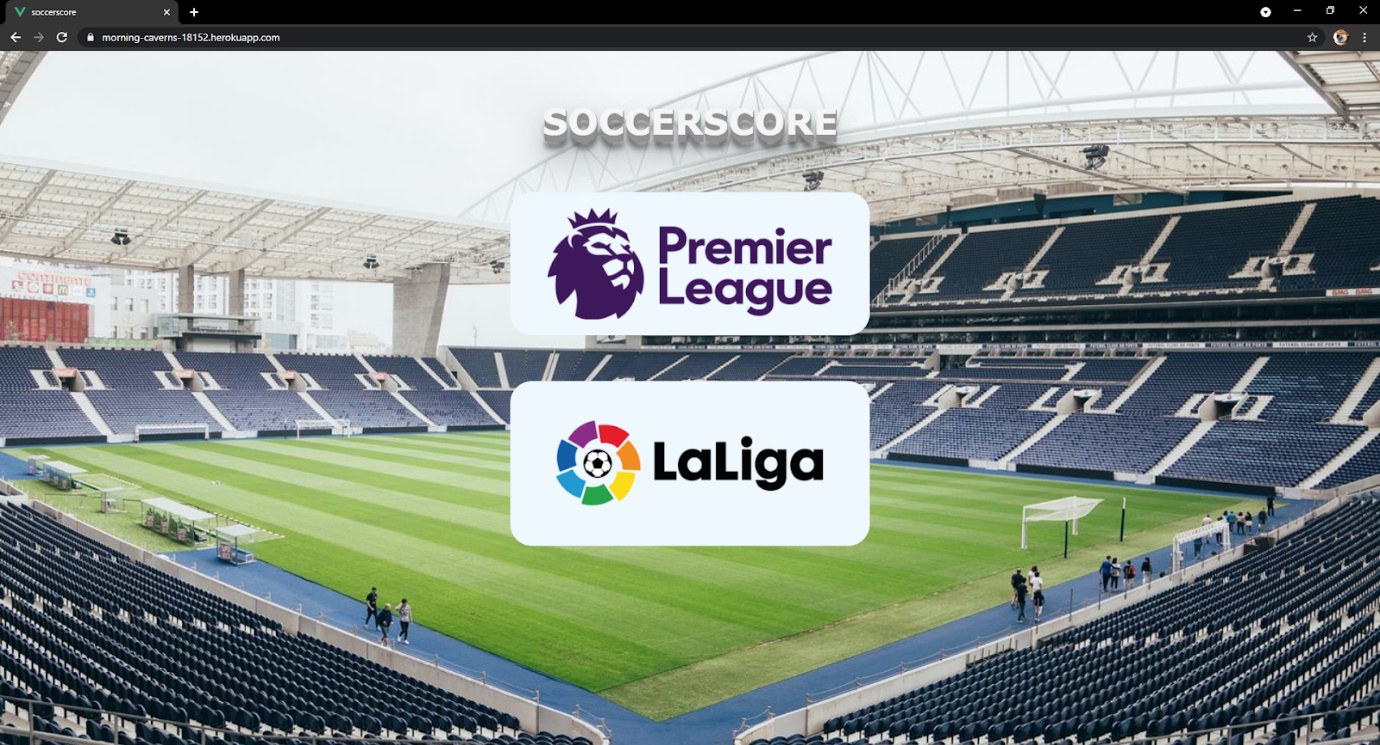
* python ./app.py



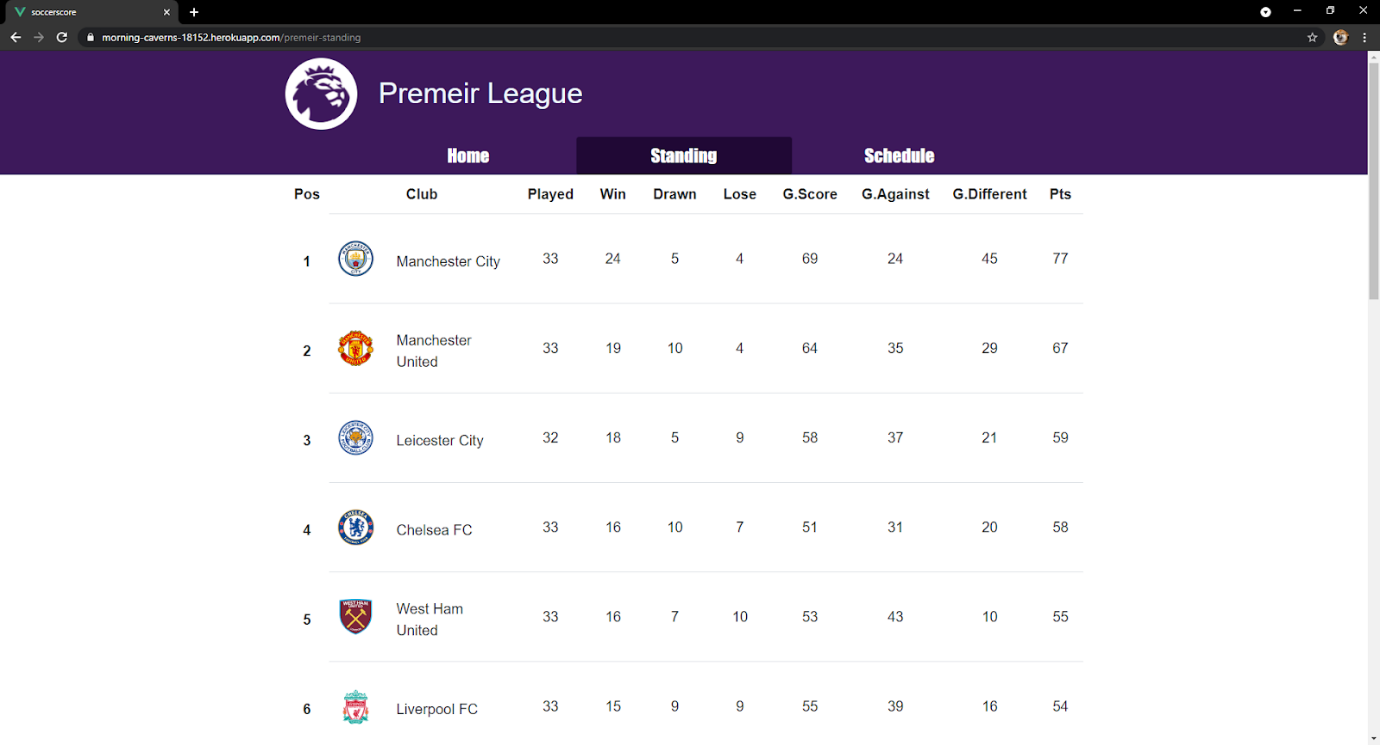
**โปรเจค SOCCERSCORE**

URL :  <https://morning-caverns-18152.herokuapp.com/>

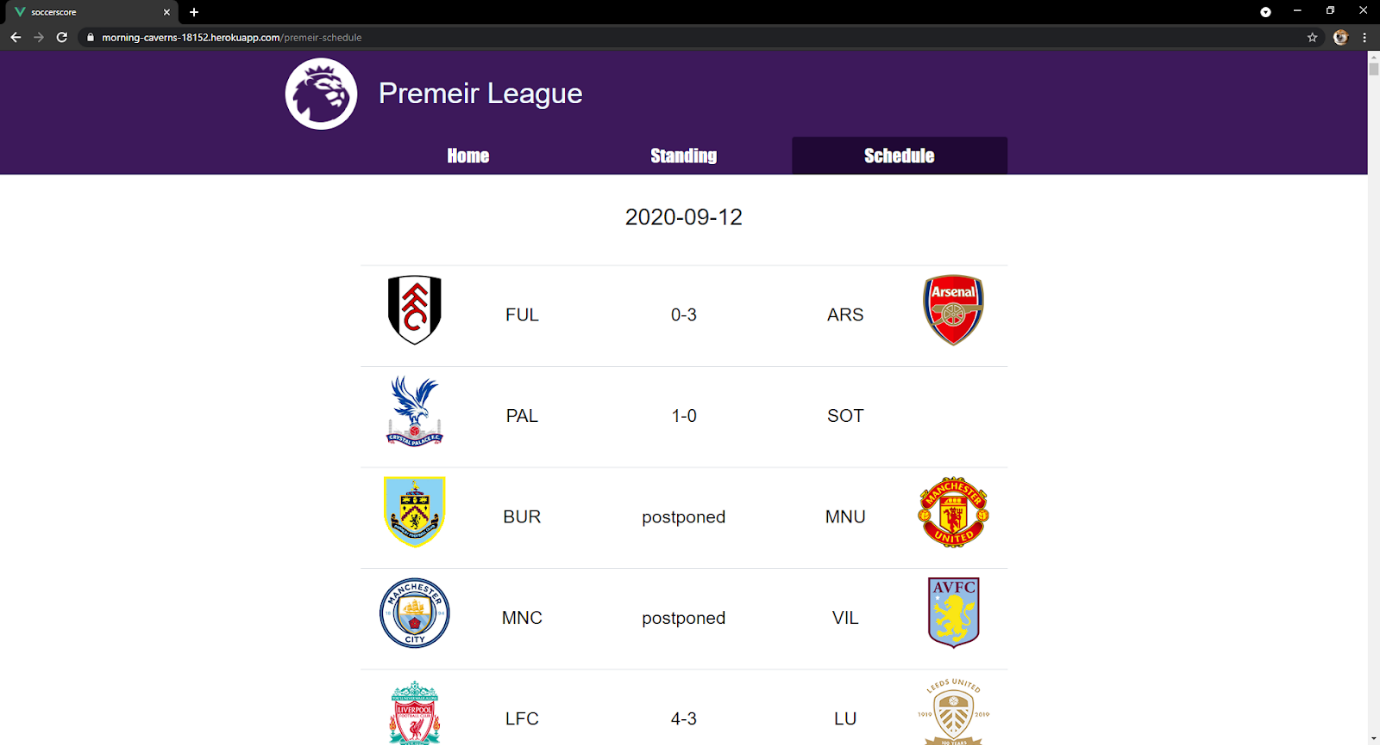
1.หน้าแรกของโปรเจค SOCCERSCORE ก็จะมีให้เลือก 2 ลีก ก็คือ Premire League และ Laliga



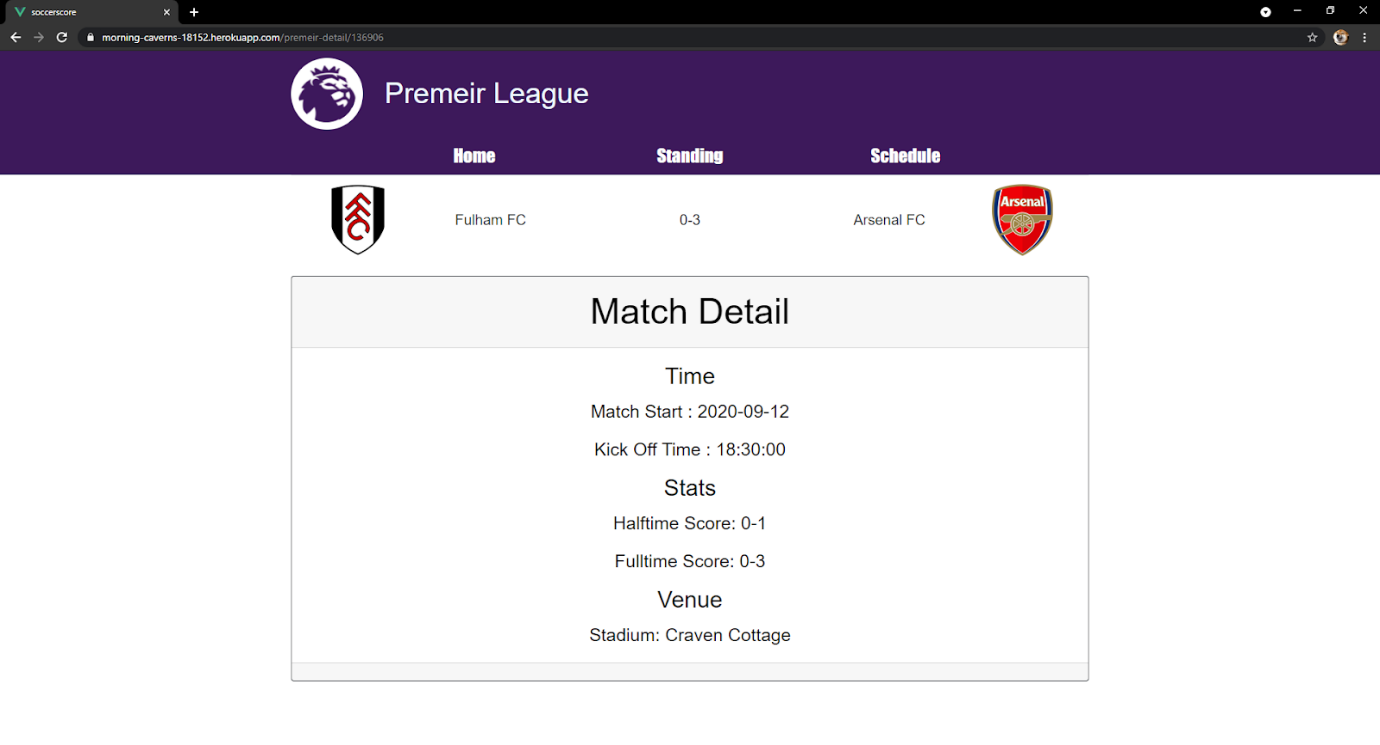
2.เมื่อคลิกเข้าไปในลีกที่เลือกเช่น Premier League หน้าแรกก็จะเป็น Standing บอกลำดับ 20 ลำดับในลีก และข้อมูลการแข่งของแต่ละทีม เช่น แพ้กี่ครั้ง ชนะกี่ครั้ง และบอกคะแนนรวมเป็นต้น



3.หน้า Schedule เป็นการบอกตารางการแข่งขันในลีกว่าวันไหนทีมไหนจะเจอคู่ไหน และสามารถเข้าไปคลิกดู Match Detail ของคู่ที่จะแข่งได้อีกด้วย



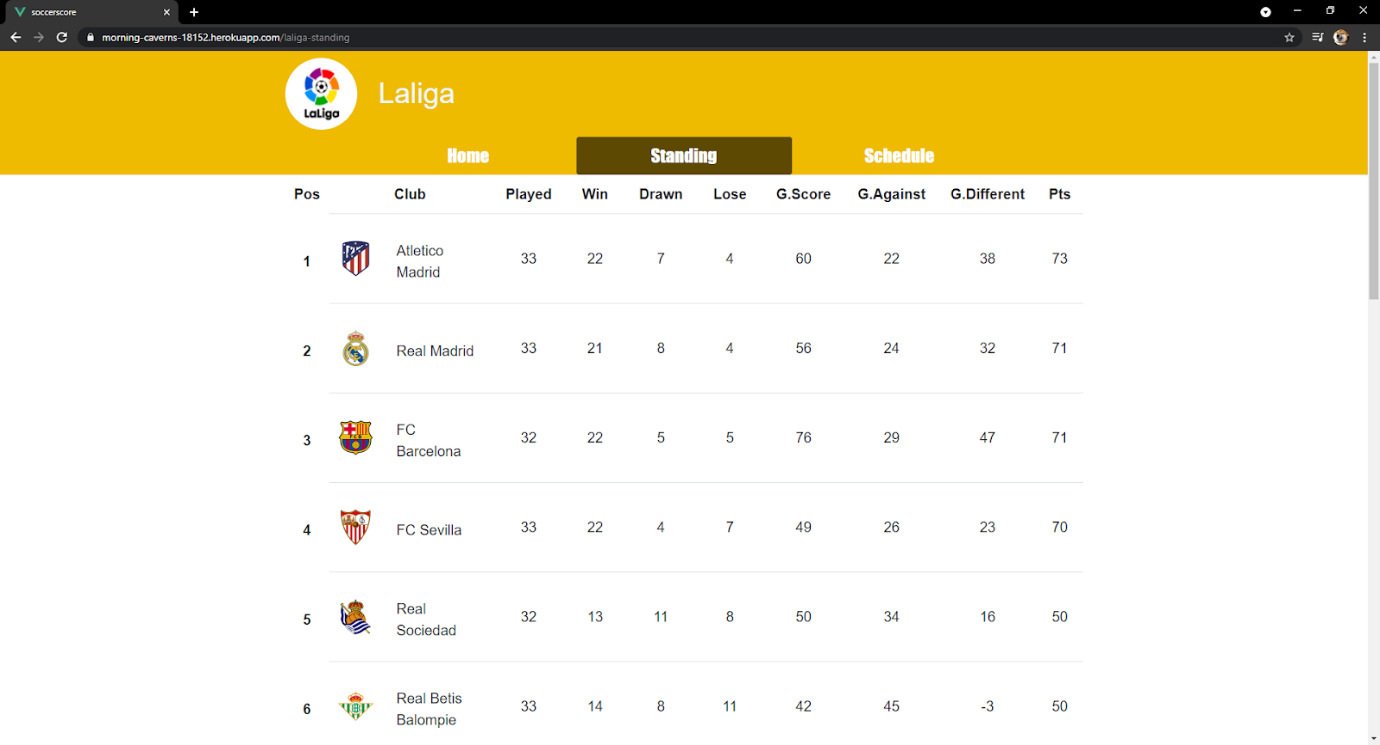
4.หน้า Match Detail บอกรายละเอียดเวลาการแข่งสถานที่ คะแนนในแต่ละ Stats



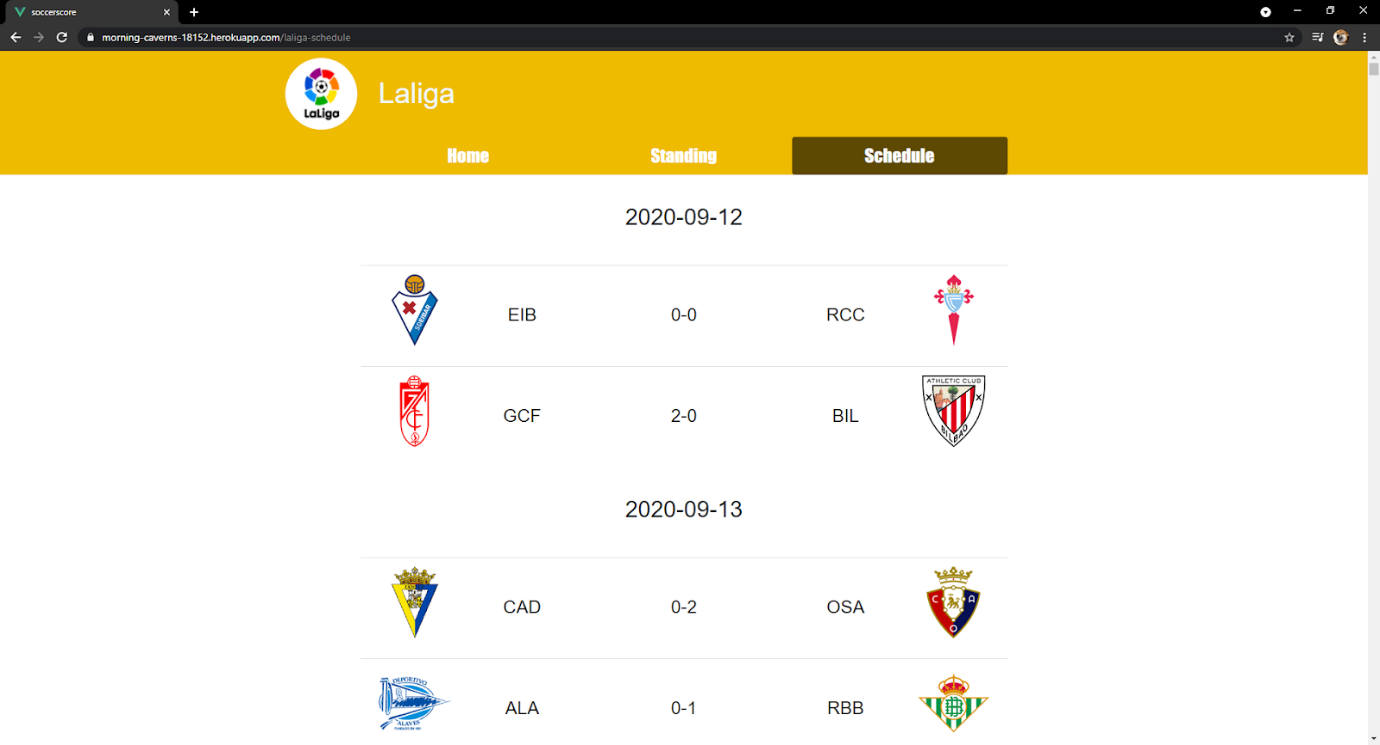
5.ถ้าต้องการกลับไปหน้าแรกให้คลิกที่ Home

**ตัวอย่างหน้าเว็บที่แสดงลีก Laliga**

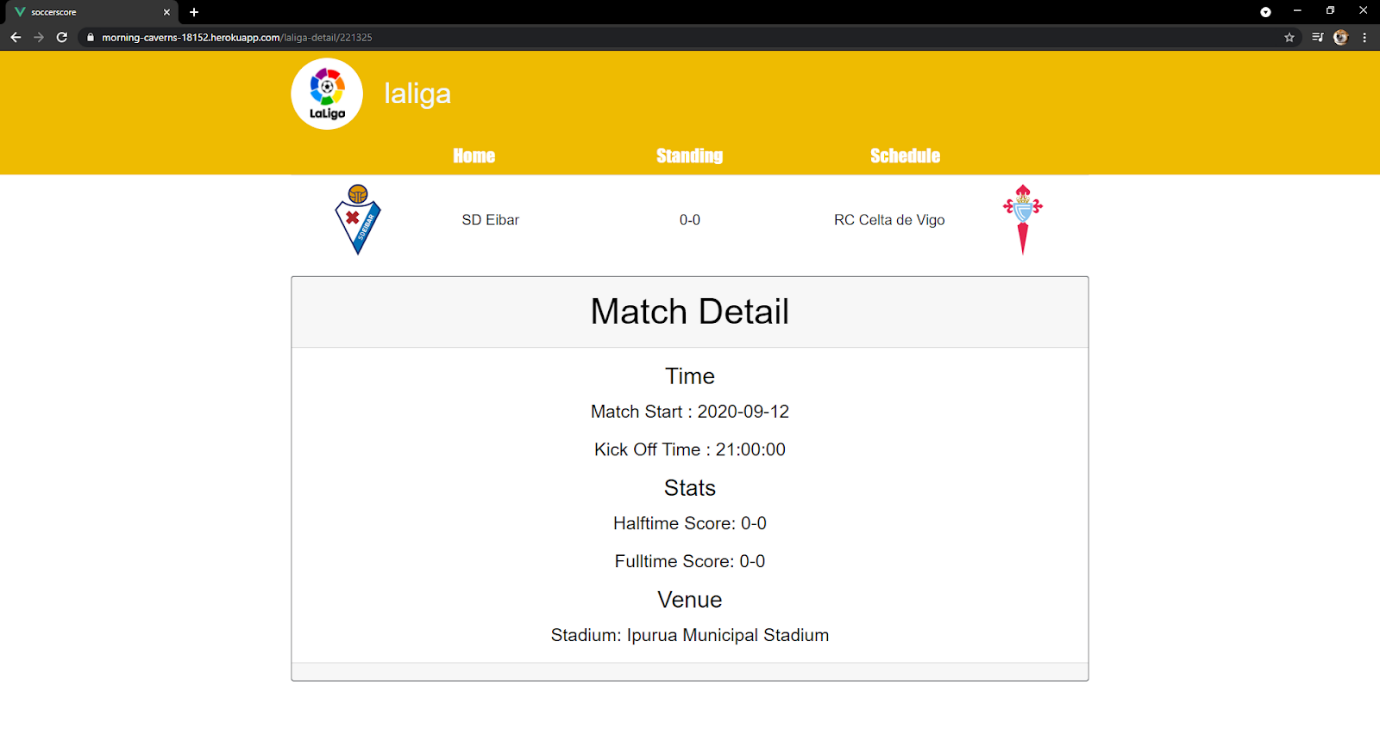
1.หน้า Standing



2.หน้า Schedule

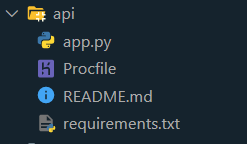


3.หน้า Match Detail



**CODE**

**ส่วน Code ของ API**



**app.py คือไฟล์ที่ทำการจัดการฐานข้อมูล และเป็น API สำหรับการส่งข้อมูลไปแสดงผลยังหน้าเว็**บ

from flask import Flask, request

from flask\_cors import CORS

from pymongo import MongoClient

from bson import json\_util

import requests

from datetime import datetime

from pytz import timezone

from apscheduler.schedulers.background import BackgroundScheduler

app = Flask(\_\_name\_\_)

cors = CORS(app, resources={r"/\*": {"origins": "\*"}})

# mongo database URI

client = MongoClient(

    "mongodb+srv://admin:3cMr8dQtNeKzROXq@cluster0.hgtii.mongodb.net"

    # "mongodb://admin:NNAppn47420@node13003-soccer-api.app.ruk-com.cloud:11027"

)

# database name

db = client["SoccerScore"]

Leagues = db.Leagues

Teams = db.Teams

Matches = db.Matches

Standings = db.Standings

headers = {"apikey": "6f889a60-8edd-11eb-9084-05c23de9546d"}

@app.route("/leagues/", methods=["GET"])

def getAllLeague():

    data = Leagues.find({})

    return json\_util.dumps({"data": data})

@app.route("/standings/<ss\_id>", methods=["GET"])

def getLeagueStandings(ss\_id):

    # current premierLeague ss = 352 and LaLiga = 1511

    seasons\_id = int(ss\_id)

    cursors = Standings.find({"season\_id": seasons\_id})

    data = list()

    # setting format the json data

    for standing\_obj in cursors:

        standings = standing\_obj["standings"]

        # print(type(standings))

        # find match  team\_data (team\_name and team\_logo) and put to array data

        for team in standings:

            team\_id = team["team\_id"]

            # find team\_data

            team\_data = Teams.find({"team\_id": team\_id})

            for item in team\_data:

                team["team\_logo"] = item["logo"]

                team["team\_name"] = item["name"]

        data = standings

    return json\_util.dumps({"data": data})

@app.route("/matches/<ss\_id>", methods=["GET"])

def getMatches(ss\_id):

    # current premierLeague ss = 352 and LaLiga = 1511

    seasons\_id = int(ss\_id)

    # query string (date\_from, date\_to)

    args = request.args

    # check date filter is exist

    # date format is Y-m=d H:M:S

    if "date\_from" in args:

        date\_from = args["date\_from"]

        # print("from", date\_from)

        if "date\_to" in args:

            date\_to = args["date\_to"]

            # print("to", date\_to)

            # if use all filter

            data = Matches.find({"season\_id": seasons\_id, "match\_start\_th": {"$gt": date\_from, "$lt": date\_to}})

            return json\_util.dumps({"data": data})

        # if use only date\_from

        data = Matches.find({"season\_id": seasons\_id, "match\_start\_th": {"$gt": date\_from}})

    else:

        # not use filter

        data = Matches.find({"season\_id": seasons\_id})

    return json\_util.dumps({"data": data})

@app.route("/match-detail/<m\_id>")

def getMatchDetail(m\_id):

    match\_id = int(m\_id)

    data = Matches.find\_one({"match\_id": match\_id})

    return json\_util.dumps({"data": data})

def prepareDB():

    # print("start update Database")

    prepareLeaguesDB()

    prepareTeamDB()

    prepareMatchesDB()

    prepareStandings()

    # print("Database update was successful. ")

    return 0

### LeaguesDB ###

# find league\_id, season\_id and country\_id in subscribLeagues

def prepareLeaguesDB():

    leagues = getSubscribLeagues()

    seasons = getLeagueSeasonInfo(leagues)

    for i in range(len(leagues)):

        # filter before put to db

        league\_data = dict()

        league\_data = leagues[i]

        league\_data["season\_data"] = seasons[i]

        filter = {"league\_id": league\_data["league\_id"]}

        update = {"$set": league\_data}

        Leagues.update\_one(filter=filter, update=update, upsert=True)

def getSubscribLeagues():

    params = {"subscribed": True}

    response = requests.get("https://app.sportdataapi.com/api/v1/soccer/leagues", headers=headers, params=params).json()

    datas = response["data"]

    return datas

def getLeagueSeasonInfo(leagues):

    seasonData = list()

    for league in leagues:

        params = {"league\_id": league["league\_id"]}

        response = requests.get(

            "https://app.sportdataapi.com/api/v1/soccer/seasons", headers=headers, params=params

        ).json()

        seasonData.append(response["data"])

    return seasonData

### TeamDB ###

# find every team in country is subscrib

def prepareTeamDB():

    for league in Leagues.find({}):

        # find team by country\_id

        params = {"country\_id": league["country\_id"]}

        response = requests.get(

            "https://app.sportdataapi.com/api/v1/soccer/teams", headers=headers, params=params

        ).json()

        # update team\_data and insert team isn't exist

        team\_data = response["data"]

        for data in team\_data:

            filter = {"team\_id": data["team\_id"]}

            update = {"$set": data}

            Teams.update\_one(filter=filter, update=update, upsert=True)

### MatchesDB ###

def prepareMatchesDB():

    current\_season\_list = getAllCurrentSeasonID()  # get current season form subscrib Leagues

    url = "https://app.sportdataapi.com/api/v1/soccer/matches"

    datas = list()

    for i in range(len(current\_season\_list)):

        current\_season = current\_season\_list[i]

        params = {"season\_id": current\_season}

        datas.append(requests.get(url, headers=headers, params=params).json()["data"])

        # sort matches data orderby date

        sortedData = sorted(datas[i], key=lambda data: data["match\_start\_iso"])

        for match in sortedData:

            filter = {"match\_id": match["match\_id"]}

            match["match\_start\_th"] = toThaiTime(match["match\_start"])

            match["match\_date\_th"] = match["match\_start\_th"][0:10]

            match["match\_time\_th"] = match["match\_start\_th"][11:]

            update = {"$set": match}

            Matches.update\_one(filter=filter, update=update, upsert=True)

def getAllCurrentSeasonID():

    current\_season\_list = list()

    for obj in Leagues.find({}):

        seasons = obj["season\_data"]

        for season in seasons:

            if season["is\_current"] == 1:

                current\_season\_list.append(season["season\_id"])

                break

    return current\_season\_list

def toThaiTime(str\_time):

    # time format

    fmt = "%Y-%m-%d %H:%M:%S"

    # convert str to datetime

    time\_obj = datetime.strptime(str\_time, fmt)

    # set default time zone

    timezone\_default = timezone("UTC")

    default\_datetime\_obj = timezone\_default.localize(time\_obj)

    # convert thai time zone

    timezone\_bkk = timezone("Asia/Bangkok")

    bkk\_datetime\_obj = default\_datetime\_obj.astimezone(timezone\_bkk)

    return bkk\_datetime\_obj.strftime(fmt)

### Standings Ranking ###

def prepareStandings():

    current\_season\_list = getAllCurrentSeasonID()

    url = "https://app.sportdataapi.com/api/v1/soccer/standings"

    for i in range(len(current\_season\_list)):

        current\_season = current\_season\_list[i]

        params = {"season\_id": current\_season}

        data = requests.get(url, headers=headers, params=params).json()["data"]

        filter = {"season\_id": data["season\_id"]}

        update = {"$set": data}

        Standings.update\_one(filter=filter, update=update, upsert=True)

if \_\_name\_\_ == "\_\_main\_\_":

    update\_schedule = BackgroundScheduler(daemon=True)

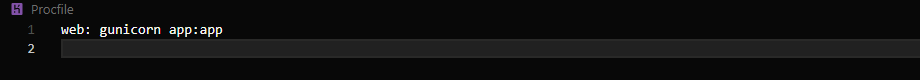
    update\_schedule.add\_job(prepareDB, "interval", hours=60)

    update\_schedule.start()

    prepareDB()

    app.run(debug=True, port=80)

**Procfile คือ Configuration File ของ Heroku**



**requirements.txt คือรายชื่อของ packet ที่จำเป็นต้องใช้ในการทำงาน**

Flask==1.1.2

pymongo[srv]

Flask\_Cors==3.0.10

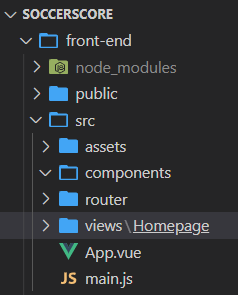
pytz==2021.1

requests==2.25.1

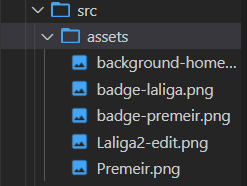
apscheduler

gunicorn

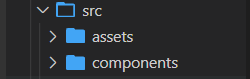
**ส่วน Code ของ Vue**



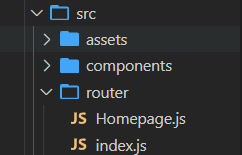
หน้าส่วนแสดงโครงสร้างทั้งหมดของโปรเจ็คโดยจะมีอยู่ 2 โฟลเดอร์ใหญ่และ 4 โฟลเดอร์แยก



ในโฟลเดอร์ย่อยของ src อันแรก จะเป็นโฟลเดอร์ assets ซึ่งเอาไว้เก็บรูปภาพต่างๆที่ใช้ในการแสดงผลของโปรแกรมทั้งหมด



ในส่วนของโฟลเดอร์ที่ 2 จะชื่อ components โดยจะเอาไว้จัดเก็บ components ต่างๆที่ได้สร้างขึ้นมา



โฟลเดอร์ที่ 3 เป็นโฟลเดอร์ router ที่จะเอาไว้ใส่ลิ้งของ components ต่างๆ Path ต่างๆ ของcomponents

**ไฟล์ Homepage.js**

export default {

  path: '/',

  component: () => import('@/views/Homepage/index.vue'),

  children: [

    {

      path: '',

      name: 'Detail',

      component: () => import('@/views/Homepage/pages/SoccerStadium.vue')

    },

    {

      path: 'premeir-standing',

      name: 'FirstMenu',

      component: () => import('@/views/Homepage/pages/premeir-standing.vue')

    },

    {

      path: 'premeir-schedule',

      name: 'premeir-schedule',

      component: () => import('@/views/Homepage/pages/premeir-schedule.vue')

    },

    {

      path: 'premeir-detail/:MatchDetail',

      name: 'premeir-detail',

      component: () => import('@/views/Homepage/pages/premeir-detail.vue'),

      props: true

    },

    {

      path: 'laliga-schedule',

      name: 'laliga-schedule',

      component: () => import('@/views/Homepage/pages/laliga-schedule.vue')

    },

    {

      path: 'laliga-standing',

      name: 'laliga-standing',

      component: () => import('@/views/Homepage/pages/laliga-standing.vue')

    },

    {

      path: 'laliga-detail/:MatchDetail',

      name: 'laliga-detail',

      component: () => import('@/views/Homepage/pages/laliga-detail.vue'),

      props: true

    }

  ]

}

**ไฟล์ index.js**

import Vue from 'vue'

import VueRouter from 'vue-router'

import Homepage from './Homepage'

Vue.use(VueRouter)

const routes = [

  Homepage

]

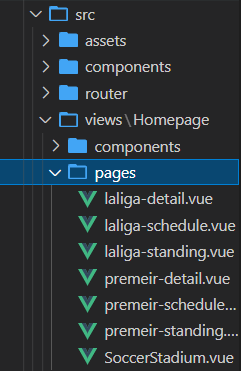
const router = new VueRouter({

  mode: 'history',

  routes

})

export default router



ในโฟลเดอร์ที่ 4 ชื่อ views เป็นโฟลเดอร์ที่ส่วนตัวเอาไว้เก็บ components ที่ใช้ โดยมี 2 โฟลเดอร์ย่อย

โดยมี components และ pages โดย pages จะเอาไว้เก็บ components ทุกตัวที่ใช้ในหน้าต่างๆ

**ไฟล์ laliga-detail.vue**

<template>

  <div>

    <!-- title img logo and Premeir league text-->

    <div class="background-titlebar">

      <div class="container">

        <div class="image">

          <img

            src="../../../assets/badge-laliga.png"

            alt=""

            class="image-position"

          />

          <h1>laliga</h1>

        </div>

      </div>

    <!-- end title img logo and Premeir league text-->

    <!-- nav menu bar -->

      <div class="navbar">

        <ul class="nav nav-pills nav-fill red">

          <li class="nav-item">

            <a class="nav-link " href="/">Home</a>

          </li>

          <li class="nav-item ">

            <a class="nav-link" href="/laliga-standing">Standing</a>

          </li>

          <li class="nav-item ">

            <a class="nav-link" href="/laliga-schedule">Schedule</a>

          </li>

        </ul>

      </div>

    <!-- end of nav menu bar  -->

    </div>

    <!-- end of title -->

    <!-- body & table -->

    <div class="container">

      <!-- match show -->

      <!-- ทำการวน loop for เพื่อดึงขอมูลมาจาก object  -->

      <div class="table table-hover table-detail">

        <table

          v-for="team in info.data"

          :key="team"

          class="list table table-striped "

        >

        <!-- กรณีที่ Match มีการเลื่อนออกไปจะไม่โชว์ score -->

          <tr v-if="team.status != 'postponed'">

            <td width="150px"><img :src="team.home\_team.logo" /></td>

            <td width="150px">{{ team.home\_team.name }}</td>

            <td width="300px">{{ team.stats.ft\_score }}</td>

            <td width="150px">{{ team.away\_team.name }}</td>

            <td width="150px"><img :src="team.away\_team.logo" /></td>

          </tr>

        </table>

      </div>

      <!-- end match show -->

      <!-- card show stat and detail -->

      <!-- ใช้ card เพื่อสร้างเป็นหน้าต่างเพื่อให้ดูสวยงาม -->

      <div class="description">

        <div class="card text-center border-dark mb-3">

          <div class="card-header">

            Match Detail

          </div>

          <div class="card-body" v-for="team in info.data" :key="team">

            <h2 class="card-title">Time</h2>

            <p class="card-text">Match Start : {{ team.match\_date\_th }}</p>

            <p class="card-text">Kick Off Time : {{ team.match\_time\_th }}</p>

            <h2 class="card-title">Stats</h2>

            <p class="card-text">Halftime Score: {{ team.stats.ht\_score }}</p>

            <p class="card-text">Fulltime Score: {{ team.stats.ft\_score }}</p>

            <h2 class="card-title">Venue</h2>

            <p class="card-text">Stadium: {{ team.venue.name }}</p>

          </div>

          <div class="card-footer text-muted"></div>

        </div>

      </div>

      <!-- end card show stat and detail -->

    </div>

    <!-- end body & table -->

  </div>

</template>

<script>

import axios from 'axios'

export default {

  data () {

    return {

      // ตัวแปรที่นำมาเก็บค่าจาก api

      info: {}

    }

  },

  // get match\_id by props

  // ใช้ function props เพื่อนำค่า params ที่ได้ส่งมาจากหน้า schedule มาใช้

  props: ['MatchDetail'],

  // get api function

  mounted () {

    // รับค่าจาก api โดยใช้ axios

    axios

    // นำตัวแปรที่ได้มาจาก props มาต่อท้ายเข้า api จะได้ข้อมูลเฉพาะของแต่ละ Match

      .get('https://soccerscoreapi.herokuapp.com/match-detail/' + this.MatchDetail)

      .then(response => {

        this.info = response

      })

  }

}

</script>

<style scoped>

.table {

  font-size: 20px;

  font-family: "B612", sans-serif;

}

a {

  font-size: 25px;

  color: white;

}

a:hover {

  background-color: #5d4901;

}

p {

  font-size: 25px;

}

td,

th {

  text-align: center;

  vertical-align: middle;

}

.nav-link {

  width: 300px;

  margin: 0;

}

.description {

  font-size: 50px;

  color: black;

  font-family: sans-serif;

}

.nav-pills .active .nav-link:not(.active) {

  background-color: #5d4901;

}

.navbar {

  margin-top: 2px;

  display: flex;

  align-items: center;

  justify-content: center;

  padding-bottom: 0px;

}

.team-logo {

  width: 30px;

  height: 30px;

}

.tab-bar {

  display: flex;

  align-items: center;

  justify-content: center;

  color: black;

  padding-left: 10px;

  padding-right: 10px;

  margin-top: 10px;

}

.container {

  width: 90vw;

  margin: 0 auto;

}

.background-titlebar {

  width: 100%;

  height: fit-content;

  background-color: #eebb01;

}

.image {

  display: flex;

  align-items: center;

  font-family: Regular, sans-serif, Georgia;

}

.image-position {

  background-color: white;

  padding: 10px;

  padding-left: 10px ;

  width: 80px;

  height: 80px;

  background: white;

  -moz-border-radius: 70px;

  -webkit-border-radius: 70px;

  border-radius: 70px;

  margin-top: 10px;

}

h1 {

  margin-left: 30px;

  color: aliceblue;

  margin-top: 15px;

}

</style>

**ไฟล์ laliga-schedule.vue**

<template>

  <div>

    <!-- title of page -->

    <div class="background">

      <!-- title img logo and Premeir league text-->

      <div class="container">

        <div class="image">

          <img

            src="../../../assets/badge-laliga.png"

            alt=""

            class="image-position"

          />

          <h1>Laliga</h1>

        </div>

      </div>

      <!-- end title img logo and Premeir league text-->

      <!-- nav menu bar -->

      <div class="navbar">

        <ul class="nav nav-pills nav-fill red">

          <li class="menu nav-item">

            <a class="nav-link " aria-current="page" href="/">Home</a>

          </li>

          <li class="menu nav-item ">

            <a class="nav-link" href="laliga-standing">Standing</a>

          </li>

          <li class="menu nav-item active">

            <a class="nav-link" href="laliga-schedule">Schedule</a>

          </li>

        </ul>

      </div>

      <!-- end nav menu bar -->

    </div>

    <!-- end title of page -->

    <!-- body & table -->

    <div class="container">

      <!-- for date of the match -->

      <!-- นำวันที่ในแต่ละ Match ออกมาแสดงโดยสร้างตัวแปรรับค่า object โดยใช้ loop for -->

      <div class="table" v-for="date in NewDate" :key="date">

        <h2>{{ date }}</h2>

      <!-- end for date of the match -->

      <!-- data in table -->

      <!-- ทำการ วน loop เพื่อดึงค่าข้อมูล Match ต่างๆมาแสดง -->

        <div v-for="team in info.data.data" :key="team">

          <table v-if="date == team.match\_date\_th" class="table  table-hover">

            <tbody>

              <!-- send match id to match-detail -->

               <!-- ใช้ router link เพื่อเป็นการส่ง params ไปยังหน้า Match Detail เพื่อ Filler เฉพาะ Match นั้นๆ -->

              <router-link

                :to="{

                  name: 'laliga-detail',

                  params: { MatchDetail: team.match\_id }

                }"

                class="match-detail"

              >

              <!-- end send match id to match-detail -->

              <!-- data in table -->

                <tr>

                  <td width="150px">

                    <img :src="team.home\_team.logo" class="img-teamlogo" />

                  </td>

                  <td width="150px" class="inrow">

                    {{ team.home\_team.short\_code }}

                  </td>

                  <!-- if statement -->

                  <!-- ในส่วนของ row นี้จะเป็นการนำสกอร์ของแต่ละ Match ออกมาแต่เนื่องจากมีบาง Match ที่เลื่อนหรือยังไม่ได้แข่งจึงใช้ if เช็คเงื่อนไขก่อนแสดงผล -->

                  <!-- กรณีแข่งขันจบแล้ว -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status == 'finished'"

                  >

                    {{ team.stats.ft\_score }}

                  </td>

                   <!-- กรณียังไม่ได้แข่งขัน -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status == 'notstarted'"

                  >

                    {{ team.match\_time\_th }}

                  </td>

                  <!-- กรณีกำลังแข่งขันอยู่หรือเลื่อนการแข่งขันออกไป -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status == 'postponed' || team.status == 'inplay'"

                  >

                    {{ team.status }}

                  </td>

                  <!-- กรณียังไม่ได้มีการประกาศเวลาแข่งขันออกมา -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status\_code == 17"

                  >

                    TBA

                  </td>

                  <!-- end if statement -->

                  <td width="150px" class="inrow">

                    {{ team.away\_team.short\_code }}

                  </td>

                  <td width="150px">

                    <img :src="team.away\_team.logo" class="img-teamlogo" />

                  </td>

                </tr>

                <!-- end of data in tale -->

              </router-link>

            </tbody>

          </table>

        </div>

      </div>

    </div>

  </div>

</template>

<script>

import axios from 'axios'

export default {

  data () {

    return {

      // ตัวแปรที่นำมาเก็บค่าจาก api

      info: {},

      // ตัวแปรที่นำมาเก็บค่าจาก api

      date: {},

      // ตัวแปรที่สร้างมาไว้สำหรับสร้าง set เพื่อกรองวันที่ ที่เหมือนกันออกเหลือ 1 วัน

      arrayMap: null,

      // นำมาเก็บค่าจาก arrayMap อีกที

      NewDate: null

      // JSON.stringify(jsArray) converts the jsArray into a string which can be stored in sessionStorage

    }

  },

  // get api function

  mounted () {

    // รับค่าจาก api โดยใช้ axios

    axios

      .get(

        'https://soccerscoreapi.herokuapp.com/matches/1511'

      )

      .then(response => {

        // รับค่าจากทั้งหมดจาก api

        this.info = response

        // รับเฉพาะ object ที่มีวันอยู่

        this.date = response.data.data

        //  ทำการ map date เข้ากับ ค่าข้างใน object ที่มีวันอยู่ โดยเก็บไว้ที่ arrayMap

        this.arrayMap = this.date.map(date => date.match\_date\_th)

        //  สร้างตัวแปรมารับค่าที่นำไปเข้า set แล้วจะได้วันที่ออกมา

        this.NewDate = new Set(this.arrayMap)

      })

  }

}

</script>

<style scoped>

.table {

  font-size: 20px;

  font-family: "B612", sans-serif;

}

.table {

  text-align: center;

  vertical-align: middle;

}

a {

  font-size: 25px;

  color: white;

}

h2 {

  padding: 40px;

  font-size: 40px bold;

}

.match-detail {

  color: black;

  text-decoration: none;

}

.inrow {

  padding-top: 50px;

}

.menu a:hover {

  background-color: #5d4901;

  text-decoration: none;

}

.nav-link {

  width: 300px;

  margin: 0;

}

.nav-pills .active .nav-link:not(.active) {

  background-color: #5d4901;

}

.navbar {

  margin-top: 2px;

  display: flex;

  align-items: center;

  justify-content: center;

  padding-bottom: 0px;

}

.match-detail {

  align-content: center;

}

.container {

  width: 90vw;

  margin: 0 auto;

}

.container table {

  display: flex;

  justify-content: center;

}

.background {

  width: 100%;

  height: fit-content;

  background-color: #eebb01;

}

.image {

  display: flex;

  align-items: center;

  font-family: Regular, sans-serif, Georgia;

}

.image-position {

  background-color: white;

  padding: 10px;

  padding-left: 10px ;

  width: 80px;

  height: 80px;

  background: white;

  -moz-border-radius: 70px;

  -webkit-border-radius: 70px;

  border-radius: 70px;

  margin-top: 10px;

}

h1 {

  margin-left: 30px;

  color: aliceblue;

  margin-top: 15px;

}

</style>

**ไฟล์ laliga-standing.vue**

<template>

  <div>

    <!-- title  -->

    <div class="background-titlebar">

      <!-- title img logo and Premeir league text-->

      <div class="container">

        <div class="image">

          <!-- ทำการเพิ่ม logo ของ League โดยใช้ <img> -->

          <img

            src="../../../assets/badge-laliga.png"

            alt=""

            class="image-position"

          />

          <h1>Laliga </h1>

        </div>

      </div>

      <!-- end title img logo and Premeir league text-->

      <!-- nav menu bar -->

      <div class="navbar">

        <!-- ทำเป็นเมนูสำหรับคลิ๊กเลือก page โดยใช้ navbar -->

        <ul class="nav nav-pills nav-fill red">

          <li class="nav-item">

            <a class="nav-link " aria-current="page" href="/">Home</a>

          </li>

          <!-- เมื่อเราอยู่ตำแหน่งหน้าไหนจะให้ปุ่ม active ที่เมนูนั้นๆ -->

          <li class="nav-item active">

            <a class="nav-link" href="laliga-standing">Standing</a>

          </li>

          <li class="nav-item">

            <a class="nav-link" href="laliga-schedule">Schedule</a>

          </li>

        </ul>

      </div>

      <!-- end of nav menu bar  -->

    </div>

    <!-- end of title -->

    <!-- body & table -->

    <div class="container">

      <!-- table -->

      <div class="table-data borderless">

        <!-- head table & colums -->

        <!-- สร้างหัวตารางตามที่ได้ออกแบบไว้ -->

        <table class="table ">

          <thead class="">

            <tr>

              <th scope="col">Pos</th>

              <th scope="col" colspan="2">Club</th>

              <th scope="col">Played</th>

              <th scope="col">Win</th>

              <th scope="col">Drawn</th>

              <th scope="col">Lose</th>

              <th scope="col">G.Score</th>

              <th scope="col">G.Against</th>

              <th scope="col">G.Different</th>

              <th scope="col">Pts</th>

            </tr>

          </thead>

        <!-- end head table & colums -->

        <!-- body data standing  -->

          <tbody>

            <!-- วน loop เพื่อดึงค่าออกจากตัวแปร object ที่ได้ get ค่าจาก api มา -->

            <tr v-for="item in info.data.data" :key="item">

              <th scope="row" class="pos-position">{{ item.position }}</th>

              <td  class="team-image">

                <img

                  :src="item.team\_logo"

                  width="50px"

                  height="50px

                "

                />

              </td>

              <td width="300px" class="team-name">{{ item.team\_name }}</td>

              <td width="180px" class="">{{ item.overall.games\_played }}</td>

              <td width="180px">{{ item.overall.won }}</td>

              <td width="180px">{{ item.overall.draw }}</td>

              <td width="180px">{{ item.overall.lost }}</td>

              <td width="180px">{{ item.overall.goals\_scored }}</td>

              <td width="180px">{{ item.overall.goals\_against }}</td>

              <td width="180px">{{ item.overall.goals\_diff }}</td>

              <td width="100">{{ item.points }}</td>

            </tr>

          </tbody>

          <!-- end body data standing  -->

        </table>

      </div>

      <!-- end table -->

    </div>

    <!-- end body & table -->

  </div>

</template>

<script>

import axios from 'axios'

export default {

  data () {

    return {

      info: {}

      // JSON.stringify(jsArray) converts the jsArray into a string which can be stored in sessionStorage

    }

  },

  mounted () {

    axios.get('https://soccerscoreapi.herokuapp.com/standings/1511').then(response => {

      this.info = response

    })

  }

}

</script>

<style scoped>

.table {

  font-size: 20px;

  font-family: "B612", sans-serif;

}

.borderless th {

  border: none;

}

a {

  font-size: 25px;

  color: white;

}

a:hover {

  background-color: #5d4901;

}

td , th{

  text-align: center;

  vertical-align: middle;

}

.nav-link {

  width: 300px;

  margin: 0;

}

.nav-pills .active .nav-link:not(.active) {

    background-color:  #5d4901;

}

.navbar {

  margin-top: 2px;

  display: flex;

  align-items: center;

  justify-content: center;

  padding-bottom: 0px;

}

.pos-position {

  padding-top: 22px;

}

.team-image {

  display: flex;

  align-items: left;

  justify-content: left;

  padding-top: 50%;

  padding-bottom: 50%;

}

.team-name {

  padding-top: 20px;

  padding-left: 20px;

  text-align: left;

}

.team-logo {

  width: 30px;

  height: 30px;

}

.tab-bar {

  display: flex;

  align-items: center;

  justify-content: center;

  color: black;

  padding-left: 10px;

  padding-right: 10px;

  margin-top: 10px;

}

.container {

  width: 90vw;

  margin: 0 auto;

}

.background-titlebar {

  width: 100%;

  height: fit-content;

  background-color: #eebb01;

}

.image {

  display: flex;

  align-items: center;

  font-family: Regular, sans-serif, Georgia;

}

.image-position {

  background-color: white;

  padding: 10px;

  padding-left: 10px ;

  width: 80px;

  height: 80px;

  background: white;

  -moz-border-radius: 70px;

  -webkit-border-radius: 70px;

  border-radius: 70px;

  margin-top: 10px;

}

h1 {

  margin-left: 30px;

  color: aliceblue;

  margin-top: 15px;

}

</style>

**ไฟล์ premeir-detail.vue**

 <template>

  <div>

    <!-- title img logo and Premeir league text-->

    <div class="background-titlebar">

      <div class="container">

        <div class="image">

          <img

            src="../../../assets/badge-premeir.png"

            alt=""

            class="image-position"

          />

          <h1>Premeir League</h1>

        </div>

      </div>

      <!-- end title img logo and Premeir league text-->

      <!-- nav menu bar -->

      <div class="navbar">

        <ul class="nav nav-pills nav-fill red">

          <li class="nav-item">

            <a class="nav-link " href="/">Home</a>

          </li>

          <li class="nav-item ">

            <a class="nav-link" href="/premeir-standing">Standing</a>

          </li>

          <li class="nav-item ">

            <a class="nav-link" href="/premeir-schedule">Schedule</a>

          </li>

        </ul>

      </div>

      <!-- end of nav menu bar  -->

    </div>

    <!-- end of title -->

    <!-- body & table -->

    <div class="container">

      <!-- match show -->

      <div class="table table-hover table-detail">\

        <!-- ทำการวน loop for เพื่อดึงขอมูลมาจาก object  -->

        <table

          v-for="team in info.data"

          :key="team"

          class="list table table-striped "

        >

        <!-- กรณีที่ Match มีการเลื่อนออกไปจะไม่โชว์ score -->

          <tr v-if="team.status != 'postponed'">

            <td width="150px"><img :src="team.home\_team.logo" /></td>

            <td width="150px">{{ team.home\_team.name }}</td>

            <td width="300px">{{ team.stats.ft\_score }}</td>

            <td width="150px">{{ team.away\_team.name }}</td>

            <td width="150px"><img :src="team.away\_team.logo" /></td>

          </tr>

        </table>

      </div>

      <!-- end match show -->

      <!-- card show stat and detail -->

      <!-- ใช้ card เพื่อสร้างเป็นหน้าต่างเพื่อให้ดูสวยงาม -->

      <div class="description">

        <div class="card text-center border-dark mb-3">

          <div class="card-header">

            Match Detail

          </div>

          <div class="card-body" v-for="team in info.data" :key="team">

            <h2 class="card-title">Time</h2>

            <p class="card-text">Match Start : {{ team.match\_date\_th }}</p>

            <p class="card-text">Kick Off Time : {{ team.match\_time\_th }}</p>

            <h2 class="card-title">Stats</h2>

            <p class="card-text">Halftime Score: {{ team.stats.ht\_score }}</p>

            <p class="card-text">Fulltime Score: {{ team.stats.ft\_score }}</p>

            <h2 class="card-title">Venue</h2>

            <p class="card-text">Stadium: {{ team.venue.name }}</p>

          </div>

          <div class="card-footer text-muted"></div>

        </div>

      </div>

      <!-- end card show stat and detail -->

    </div>

    <!-- end body & table -->

  </div>

</template>

<script>

import axios from 'axios'

export default {

  data () {

    return {

      // ตัวแปรที่นำมาเก็บค่าจาก api

      info: {}

    }

  },

  // get match\_id by props

  // ใช้ function props เพื่อนำค่า params ที่ได้ส่งมาจากหน้า schedule มาใช้

  props: ['MatchDetail'],

  // get api function

  mounted () {

    // รับค่าจาก api โดยใช้ axios

    axios

    // นำตัวแปรที่ได้มาจาก props มาต่อท้ายเข้า api จะได้ข้อมูลเฉพาะของแต่ละ Match

      .get('https://soccerscoreapi.herokuapp.com/match-detail/' + this.MatchDetail)

      .then(response => {

        this.info = response

      })

  }

}

</script>

<style scoped>

.table {

  font-size: 20px;

  font-family: "B612", sans-serif;

}

a {

  font-size: 25px;

  color: white;

}

a:hover {

  background-color: #200835;

}

p {

  font-size: 25px;

}

td,

th {

  text-align: center;

  vertical-align: middle;

}

.nav-link {

  width: 300px;

  margin: 0;

}

.description {

  font-size: 50px;

  color: black;

  font-family: sans-serif;

}

.nav-pills .active .nav-link:not(.active) {

  background-color: #200835;

}

.navbar {

  margin-top: 2px;

  display: flex;

  align-items: center;

  justify-content: center;

  padding-bottom: 0px;

}

.team-logo {

  width: 30px;

  height: 30px;

}

.tab-bar {

  display: flex;

  align-items: center;

  justify-content: center;

  color: black;

  padding-left: 10px;

  padding-right: 10px;

  margin-top: 10px;

}

.container {

  width: 90vw;

  margin: 0 auto;

}

.background-titlebar {

  width: 100%;

  height: fit-content;

  background-color: #3d195b;

}

.image {

  display: flex;

  align-items: center;

  font-family: Regular, sans-serif, Georgia;

}

.image-position {

  background-color: white;

  padding: 10px 10px;

  width: 100px;

  height: 100px;

  background: white;

  -moz-border-radius: 70px;

  -webkit-border-radius: 70px;

  border-radius: 70px;

  margin-top: 10px;

}

h1 {

  margin-left: 30px;

  color: aliceblue;

  margin-top: 15px;

}

</style>

**ไฟล์ premeir-schedule.vue**

<template>

  <div>

    <!-- title of page -->

    <div class="background">

      <!-- title img logo and Premeir league text-->

      <div class="container">

        <div class="image">

          <img

            src="../../../assets/badge-premeir.png"

            alt=""

            class="image-position"

          />

          <h1>Premeir League</h1>

        </div>

      </div>

      <!-- end title img logo and Premeir league text-->

      <!-- nav menu bar -->

      <div class="navbar">

        <ul class="nav nav-pills nav-fill red">

          <li class="menu nav-item">

            <a class="nav-link " aria-current="page" href="/">Home</a>

          </li>

          <li class="menu nav-item ">

            <a class="nav-link" href="premeir-standing">Standing</a>

          </li>

          <li class="menu nav-item active">

            <a class="nav-link" href="premeir-schedule">Schedule</a>

          </li>

        </ul>

      </div>

      <!-- end nav menu bar -->

    </div>

    <!-- end title of page -->

    <!-- body & table -->

    <div class="container">

      <!-- for date of the match -->

      <!-- นำวันที่ในแต่ละ Match ออกมาแสดงโดยสร้างตัวแปรรับค่า object โดยใช้ loop for -->

      <div class="table" v-for="date in NewDate" :key="date">

        <h2>{{ date }}</h2>

      <!-- end for date of the match -->

      <!-- data in table -->

      <!-- ทำการ วน loop เพื่อดึงค่าข้อมูล Match ต่างๆมาแสดง -->

        <div v-for="team in info.data.data" :key="team">

          <table v-if="date == team.match\_date\_th" class="table  table-hover">

            <tbody>

              <!-- send match id to match-detail -->

              <!-- ใช้ router link เพื่อเป็นการส่ง params ไปยังหน้า Match Detail เพื่อ Filler เฉพาะ Match นั้นๆ -->

              <router-link

                :to="{

                  name: 'premeir-detail',

                  params: { MatchDetail: team.match\_id }

                }"

                class="match-detail"

              >

              <!-- end send match id to match-detail -->

              <!-- data in table -->

                <tr>

                  <td width="150px">

                    <img :src="team.home\_team.logo" class="img-teamlogo" />

                  </td>

                  <td width="150px" class="inrow">

                    {{ team.home\_team.short\_code }}

                  </td>

                  <!-- if statement -->

                  <!-- ในส่วนของ row นี้จะเป็นการนำสกอร์ของแต่ละ Match ออกมาแต่เนื่องจากมีบาง Match ที่เลื่อนหรือยังไม่ได้แข่งจึงใช้ if เช็คเงื่อนไขก่อนแสดงผล -->

                  <!-- กรณีแข่งขันจบแล้ว -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status == 'finished'"

                  >

                    {{ team.stats.ft\_score }}

                  </td>

                  <!-- กรณียังไม่ได้แข่งขัน -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status == 'notstarted'"

                  >

                    {{ team.match\_time\_th }}

                  </td>

                  <!-- กรณีกำลังแข่งขันอยู่หรือเลื่อนการแข่งขันออกไป -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status == 'postponed' || team.status == 'inplay'"

                  >

                    {{ team.status }}

                  </td>

                  <!-- กรณียังไม่ได้มีการประกาศเวลาแข่งขันออกมา -->

                  <td

                    width="300px"

                    class="inrow"

                    v-if="team.status\_code == 17"

                  >

                    TBA

                  </td>

                  <!-- end if statement -->

                  <td width="150px" class="inrow">

                    {{ team.away\_team.short\_code }}

                  </td>

                  <td width="150px">

                    <img :src="team.away\_team.logo" class="img-teamlogo" />

                  </td>

                </tr>

                <!-- end of data in tale -->

              </router-link>

            </tbody>

          </table>

        </div>

        <!-- end of data in table -->

      </div>

    </div>

    <!-- end of body & table -->

  </div>

</template>

<script>

import axios from 'axios'

export default {

  data () {

    return {

      // ตัวแปรที่นำมาเก็บค่าจาก api

      info: {},

     // ตัวแปรที่นำมาเก็บค่าจาก api

      date: {},

      // ตัวแปรที่สร้างมาไว้สำหรับสร้าง set เพื่อกรองวันที่ ที่เหมือนกันออกเหลือ 1 วัน

      arrayMap: null,

      // นำมาเก็บค่าจาก arrayMap อีกที

      NewDate: null

      // JSON.stringify(jsArray) converts the jsArray into a string which can be stored in sessionStorage

    }

  },

  // get api function

  mounted () {

    // รับค่าจาก api โดยใช้ axios

    axios

      .get(

        'https://soccerscoreapi.herokuapp.com/matches/352'

      )

      .then(response => {

        // รับค่าจากทั้งหมดจาก api

        this.info = response

        // รับเฉพาะ object ที่มีวันอยู่

        this.date = response.data.data

        //  ทำการ map date เข้ากับ ค่าข้างใน object ที่มีวันอยู่ โดยเก็บไว้ที่ arrayMap

        this.arrayMap = this.date.map(date => date.match\_date\_th)

        //  สร้างตัวแปรมารับค่าที่นำไปเข้า set แล้วจะได้วันที่ออกมา

        this.NewDate = new Set(this.arrayMap)

      })

  }

}

</script>

<style scoped>

.table {

  font-size: 20px;

  font-family: "B612", sans-serif;

}

/\* set data in table to center \*/

.table {

  text-align: center;

  vertical-align: middle;

}

a {

  font-size: 25px;

  color: white;

}

/\* date of match \*/

h2 {

  padding: 40px;

  font-size: 40px bold;

}

.match-detail {

  color: black;

  text-decoration: none;

}

.inrow {

  padding-top: 50px;

}

.menu a:hover {

  background-color: #200835;

  text-decoration: none;

}

.nav-link {

  width: 300px;

  margin: 0;

}

.nav-pills .active .nav-link:not(.active) {

  background-color: #200835;

}

.navbar {

  margin-top: 2px;

  display: flex;

  align-items: center;

  justify-content: center;

  padding-bottom: 0px;

}

.match-detail {

  align-content: center;

}

.container {

  width: 90vw;

  margin: 0 auto;

}

.container table {

  display: flex;

  justify-content: center;

}

.background {

  width: 100%;

  height: fit-content;

  background-color: #3d195b;

}

.image {

  display: flex;

  align-items: center;

  font-family: Regular, sans-serif, Georgia;

}

.image-position {

  background-color: white;

  padding: 10px 10px;

  width: 100px;

  height: 100px;

  background: white;

  -moz-border-radius: 70px;

  -webkit-border-radius: 70px;

  border-radius: 70px;

  margin-top: 10px;

}

h1 {

  margin-left: 30px;

  color: aliceblue;

  margin-top: 15px;

}

</style>

**ไฟล์ premeir-standing.vue**

<template>

  <div>

    <!-- title  -->

    <div class="background-titlebar">

      <!-- title img logo and Premeir league text-->

      <div class="container">

        <div class="image">

          <!-- ทำการเพิ่ม logo ของ League โดยใช้ <img> -->

          <img

            src="../../../assets/badge-premeir.png"

            alt=""

            class="image-position"

          />

          <h1>Premeir League</h1>

        </div>

      </div>

      <!-- end title img logo and Premeir league text-->

      <!-- nav menu bar -->

      <div class="navbar">

        <!-- ทำเป็นเมนูสำหรับคลิ๊กเลือก page โดยใช้ navbar -->

        <ul class="nav nav-pills nav-fill red">

          <li class="nav-item">

            <a class="nav-link " aria-current="page" href="/">Home</a>

          </li>

          <!-- เมื่อเราอยู่ตำแหน่งหน้าไหนจะให้ปุ่ม active ที่เมนูนั้นๆ -->

          <li class="nav-item active">

            <a class="nav-link" href="premeir-standing">Standing</a>

          </li>

          <li class="nav-item">

            <a class="nav-link" href="premeir-schedule">Schedule</a>

          </li>

        </ul>

      </div>

      <!-- end of nav menu bar  -->

    </div>

    <!-- end of title -->

    <!-- body & table -->

    <div class="container">

      <!-- table -->

      <div class="table-data borderless">

        <!-- head table & colums -->

        <!-- สร้างหัวตารางตามที่ได้ออกแบบไว้ -->

        <table class="table ">

          <thead class="">

            <tr>

              <th scope="col">Pos</th>

              <th scope="col" colspan="2">Club</th>

              <th scope="col">Played</th>

              <th scope="col">Win</th>

              <th scope="col">Drawn</th>

              <th scope="col">Lose</th>

              <th scope="col">G.Score</th>

              <th scope="col">G.Against</th>

              <th scope="col">G.Different</th>

              <th scope="col">Pts</th>

            </tr>

          </thead>

          <!-- end head table & colums -->

          <!-- body data standing  -->

          <tbody>

            <!-- วน loop เพื่อดึงค่าออกจากตัวแปร object ที่ได้ get ค่าจาก api มา -->

            <tr v-for="item in info.data.data" :key="item">

              <!-- นำข้อมูลที่ได้มาแสดงโดยใช้ตัวแปรทีไ่ด้สร้างขึ้นมาเรียก object  -->

              <th scope="row" class="pos-position">{{ item.position }}</th>

              <td  class="team-image">

                <img

                  :src="item.team\_logo"

                  width="50px"

                  height="50px

                "

                />

              </td>

              <td width="300px" class="team-name">{{ item.team\_name }}</td>

              <td width="180px" class="">{{ item.overall.games\_played }}</td>

              <td width="180px">{{ item.overall.won }}</td>

              <td width="180px">{{ item.overall.draw }}</td>

              <td width="180px">{{ item.overall.lost }}</td>

              <td width="180px">{{ item.overall.goals\_scored }}</td>

              <td width="180px">{{ item.overall.goals\_against }}</td>

              <td width="180px">{{ item.overall.goals\_diff }}</td>

              <td width="100">{{ item.points }}</td>

            </tr>

          </tbody>

          <!-- end body data standing  -->

        </table>

      </div>

      <!-- end table -->

    </div>

    <!-- end body & table -->

  </div>

</template>

<script>

import axios from 'axios'

export default {

  data () {

    return {

      info: {}

      // JSON.stringify(jsArray) converts the jsArray into a string which can be stored in sessionStorage

    }

  },

  // get api function

  // ใช้ axios เพื่อ get ค่าจาก api ของ backend

  mounted () {

    axios.get('https://soccerscoreapi.herokuapp.com/standings/352').then(response => {

      // สร้างตัวแปร object มาเพื่อเก็บค่า api

      this.info = response

    })

  }

}

</script>

<style scoped>

.table {

  font-size: 20px;

  font-family: "B612", sans-serif;

}

/\* make table borderless \*/

.borderless th {

  border: none;

}

a {

  font-size: 25px;

  color: white;

}

/\* hover of nav bar color \*/

a:hover {

  background-color: #200835;

}

/\* make text in row go center \*/

td , th{

  text-align: center;

  vertical-align: middle;

}

/\* size of nav menu \*/

.nav-link {

  width: 300px;

  margin: 0;

}

/\* color of active nav menu \*/

.nav-pills .active .nav-link:not(.active) {

    background-color:  #200835;

}

/\* set position of nav menu \*/

.navbar {

  margin-top: 2px;

  display: flex;

  align-items: center;

  justify-content: center;

  padding-bottom: 0px;

}

/\* set 1st colume to center \*/

.pos-position {

  padding-top: 22px;

}

/\* set 2nd to center \*/

.team-image {

  display: flex;

  align-items: left;

  justify-content: left;

  padding-top: 50%;

  padding-bottom: 50%;

}

.team-name {

  padding-top: 20px;

  padding-left: 20px;

  text-align: left;

}

.team-logo {

  width: 30px;

  height: 30px;

}

.tab-bar {

  display: flex;

  align-items: center;

  justify-content: center;

  color: black;

  padding-left: 10px;

  padding-right: 10px;

  margin-top: 10px;

}

.container {

  width: 90vw;

  margin: 0 auto;

}

.background-titlebar {

  width: 100%;

  height: fit-content;

  background-color: #3d195b;

}

.image {

  display: flex;

  align-items: center;

  font-family: Regular, sans-serif, Georgia;

}

.image-position {

  background-color: white;

  padding: 10px 10px;

  width: 100px;

  height: 100px;

  background: white;

  -moz-border-radius: 70px;

  -webkit-border-radius: 70px;

  border-radius: 70px;

  margin-top: 10px;

}

h1 {

  margin-left: 30px;

  color: aliceblue;

  margin-top: 15px;

}

</style>

**ไฟล์ SoccerStadium.vue**

<template>

  <!-- title and background -->

  <div class="vuetify">

    <!-- สร้าง Text กราฟฟิคโดยใช้ CSS -->

    <div class="plate">

      <p class="text">SOCCERSCORE</p>

    </div>

    <!-- end title and background -->

    <!-- img menu -->

    <div class="ct-center">

      <div class="row">

        <!-- ทำการลิ้ง button โดยใช้ <a> -->

        <a href="premeir-standing" class="btn space">

          <!-- เพิ่มรูปภาพโดยใช้ <img> -->

          <img src="../../../assets/Premeir.png" class="image-home-premeir " />

        </a>

      </div>

      <div class="row">

        <!-- ทำการลิ้ง button โดยใช้ <a> -->

        <a href="laliga-standing" class="btn">

          <!-- เพิ่มรูปภาพโดยใช้ <img> -->

          <img

            src="../../../assets/Laliga2-edit.png"

            class="image-home-laliga "

          />

        </a>

      </div>

    </div>

    <!-- end img menu -->

  </div>

</template>

<script>

export default {

  data () {

    return {}

  }

}

</script>

<style scoped>

#home {

  align-content: center;

  size: 60;

}

body {

  background-color: #c4c4c4;

}

/\* grahfic title \*/

.text {

  position: absolute;

  top: 10%;

  right: 50%;

  transform: translate(50%, -50%);

  text-transform: uppercase;

  font-family: verdana;

  font-size: 50px;

  font-weight: 700;

  color: #f5f5f5;

  text-shadow: 1px 1px 1px #919191, 1px 2px 1px #919191, 1px 3px 1px #919191,

    1px 4px 1px #919191, 1px 5px 1px #919191, 1px 6px 1px #919191,

    1px 7px 1px #919191, 1px 8px 1px #919191, 1px 9px 1px #919191,

    1px 10px 1px #919191, 1px 18px 6px rgba(16, 16, 16, 0.4),

    1px 22px 10px rgba(16, 16, 16, 0.2), 1px 25px 35px rgba(16, 16, 16, 0.2),

    1px 30px 60px rgba(16, 16, 16, 0.4);

}

/\* set img menu to center \*/

.ct-center {

  display: flex;

  justify-content: center;

  flex-direction: column;

  align-items: center;

}

.space {

  margin-top: 50px;

  margin-bottom: 50px;

}

.image-home-premeir {

  width: 500px;

  height: 200px;

  opacity: 1;

  border: 2px solid transparent;

  border-radius: 30px;

  background-color: aliceblue;

  padding: 20px 50px;

  margin-top: 140px;

}

.image-home-laliga {

  width: 500px;

  height: 230px;

  opacity: 1;

  border: 2px solid transparent;

  border-radius: 30px;

  background-color: aliceblue;

  background-position: center;

  background-size: cover;

  padding: 20px 50px;

}

/\* background position \*/

.vuetify {

  background-image: url(../../../assets/background-homepage.jpg);

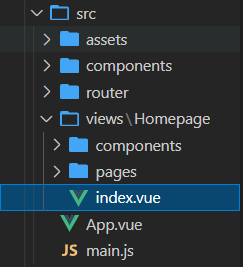
  background-size: cover;

  height: 100vh;

  background-position: center;

}

</style>



มีอีก 1 ไฟล์ที่อยู่ใน Homepage คือ index.vue ใช้สำหรับการ router ในส่วนของ router

**ไฟล์ index.vue**

<template>

   <section id="Homepage">

      <router-view />

      <head>

        <H1>SOCCERSCORE</H1>

      </head>

   </section>

</template>

App.vue ก็เป็นส่วน router view เช่นกัน และส่วนสุดท้ายก็คือ main.js เป็นส่วนการ render view

**ไฟล์ App.vue**

<template>

   <section id="Homepage">

      <router-view />

      <head>

        <H1>SOCCERSCORE</H1>

      </head>

   </section>

</template>

**ข้อมูลของ API**

ข้อมูลที่เก็บในฐานข้อมูลดึงมาจาก <https://app.sportdataapi.com/> โดยจะดึงมาทุกๆ 1 ชั่วโมงเพื่ออัพเดทฐานข้อมูล โดยในส่วนของ api ฟรีนั้นสามารถดึงข้อมูลได้ 2 ลีค และดึงได้เพียง 500 request ต่อสัปดาห์ ซึ่งสามารถเพิ่มจำนวนของ request และจำนวนของลีคได้ ขึ้นอยู่กับ Subscribe ที่เลือกไว้