Brain Radiogenomic System

Readme File

Table of Contents

[Flask Serve: 2](#_Toc122866019)

[Requirements: 2](#_Toc122866020)

[File Directory: 2](#_Toc122866021)

[React Server: 3](#_Toc122866022)

[Requirements: 3](#_Toc122866023)

[File Directory: 4](#_Toc122866024)

[Node Server: 5](#_Toc122866025)

[Requirements: 5](#_Toc122866026)

[File Directory: 5](#_Toc122866027)

[Socket Server: 6](#_Toc122866028)

[Requirements: 6](#_Toc122866029)

[File Directory: 6](#_Toc122866030)

[AI Models Information: 6](#_Toc122866031)

[Unet-segmentation Model: 6](#_Toc122866032)

[Radiomic Features: 6](#_Toc122866033)

[IDH Model: 7](#_Toc122866034)

[TCP53 Model: 7](#_Toc122866035)

# Flask Serve:

## Requirements:

Run “pip install dependencies.txt” to install packages.

|  |  |
| --- | --- |
| **Packages** | **Version** |
| Python | 3.10 |
| TensorFlow | 2.10.0 |
| Flask | 2.2.2 |
| Flask-Cors | 3.0.10 |
| nibabel | 4.0.2 |
| pyradiomics | 3.0.1 |
| SimpleITK | 2.2.0 |
| scikit-learn | 1.1.3 |
| matplotlib | 3.6.2 |
| Pickle | 4.0 |
| pybase64 | 1.2.3 |

## File Directory:

Flask

Venv

jpg Converted

Models

Upload

App.js

In ***Flask*** folder entire files of flask server placed.

***Vene*** folder contain three folders to know:

1. ***jpg Converted:***

This directory holds all modalities scans which is upload by user in JPG format.

1. ***Models:***

This directory hold all trained models used for predictions.

* **tNewClast.h5** model is used for Segmentation of medical images.
* **IDH\_Prediction\_model.h5** is used to predict IDH mutation.
* **TCP53\_Prediction\_model.h5** is used to predict TCP53 mutation.
* **Model1.pkl** is used to predict survival rate.

1. ***Upload:***

This directory hold all four modalities scans uploaded by user.

1. ***App.js:***

This file contains all Routes having logical implementation of flask APIs.

* **@app.route(“/api/data”)** is receiving Scans uploaded by user from React server and save all four scans into **Upload** folder, then convertscans into jpg format and save into **jpg Converted** folder, then send jpg images of scans to user side.
* **@app.route(“/Segmentation”)** is performing all functionalities related to segmentation of tumor. First segmentation model is loaded then images is loaded from upload folder which is first preprocessed, converted to Two dimensional array and stack all scans for segmentation. After model done segmentation segmented imaged and original image is send back to user.
* **@app.route(“/features”)** is preforming all functionalities related to features extractions, IDH prediction, TCP53 prediction and survival rate prediction. First flair modality and segmented image is load for features extraction, after features extraction these features is given as a feed for prediction to IDH, TCP53 and survival rate models. All predictions result and some specific features are send back to user.

# React Server:

## Requirements:

To install all these packages run “npm install” in command prompt of client directory.

|  |  |
| --- | --- |
| **Packages** | **Version** |
| Bootstrap | 5.2.3 |
| Char.Js | 3.8.0 |
| jspdf | 2.5.1 |
| Redux | 4.2.0 |
| Redux-devtoos-extension | 2.13.9 |
| Simple-peer | 9.11.1 |
| Markerjs2 | 2.28.0 |
| React-dom | 8.0.2 |
| React-router-dom | 6.3.0 |
| Socket-io-client | 4.5.2 |
| react-material-file-upload | 0.0.4 |
| React-magnifier | 3.0.4 |
| React-toastify | 9.0.8 |
| react-zoom-pan-pinch | 2.1.3 |

## File Directory:

Client

Src

adminRoutes

patientRoutes

doctorRoutes

radiologistRoutes

receptionistRoutes

components

Redux.

App.js

In **Client** directory all the files of flask server exist related to project.

In **Src** directory folder of each user exist:

1. **AdminRoutes:**

In this directory all components and functionalities related to admin exists. like

Register users, view users, edit users, delete users and view reports etc.

1. **PatientRoutes:**

In this directory all components and functionalities related to patient exists. Like

View profile, edit profile, view assigned doctor, chat and call with doctor, view Scans, view reports and view **RX**/**REC** (Prescriptions/Recommendations) history.

1. **DoctorRoutes:**

In this directory all components and functionalities related to doctor exists. Like

View Profile, edit profile, view assigned patients, chat and call with patient, view Scans and reports, assign radiologist to patients and do **RX & REC**.

1. **RadiologistRoutes:**

In this directory all components and functionalities related to radiologist exists. Like view profile, edit profile, view assigned patient, upload MRI Scans and generate AI reports etc.

1. **ReceptionistRoutes:**

In this directory all components related to receptionist exists. Like view profile, edit profile, register patient, add symptoms and assign doctor to patient.

1. **Components:**

In this directory all components of project exist. Like charts, sidebars etc.

1. **Redux:**

In this director all the functionalities of global state management is done.

**App.js:**

In app.js file all the routing of project is done.

# Node Server:

## Requirements:

|  |  |
| --- | --- |
| **Packages** | **Versions** |
| bcrypt | 5.0.1 |
| cookie-parser | 1.4.4 |
| cors | 2.8.5 |
| jsonwebtoken | 8.5.1 |
| dotenv | 16.0.0 |
| express | 4.16.1 |
| mongoose | 6.3.0 |
| nodemailer | 6.7.5 |
| nodemon | 2.0.15 |
| qr-image | 3.2.0 |
| qrcode | 1.5.1 |

## File Directory:

Server

App\_server

Routes

Controller

Middlewares

Models

App.js

In **Server** directory all the files related to node server exist.

In **App\_Server** directory one folder routes exist:

1. **Routes:**

This folder contains files relates to each users having its end-points or routes like Admin routes of GET, POST or DELETE and other users.

In **Controller** directory each user has its own files where its controls are implemented to do all logical operations of project.

In **Middleware** directory each user has its own files where its authentication is implemented.

In **Models** directory each user has its own files where its Schemas are implemented.

In **App.js** file all routing of node server is done.

# Socket Server:

## Requirements:

|  |  |
| --- | --- |
| **Packages** | **Version** |
| socket.io | 4.5.2 |
| nodemon | 2.0.20 |

## File Directory:

Socket

Index.js

In **Index.js** all events and functionalities to chat and video call is implanted.

It has Events.

1. Send message.
2. Receive message.
3. Call user.
4. Answer call.

# AI Models Information:

## Unet-segmentation Model:

In project, we use Unet channel last model for segmentation of tumor.

1. First load images from directory.
2. Convert into ndarray.
3. Convet 3D ndarrays to 2D ndarray.
4. Stack all four modulaties to axis 3 or channel last.
5. All prepared images feed to model for training.
6. Now Model is capable of prediction on CPU or GPU.

## Radiomic Features:

In project, we use PyRadiomic library for features extraction.

1. First get segmented tumor from Unet model.
2. Load flair modality and segmented image.
3. Construct image from array using SimpleITK library.
4. Extract features of images by passing into functions implemented to extract features and generate Dataframe.

## IDH Model:

Sequential CNN model is used for the prediction of IDH marker.

## TCP53 Model:

Sequential CNN model is used for the prediction of TCP53 marker.