**Application Deployment in AWS**

Application needed to be hosted in AWS

1. Annotation Tool
2. Admin App
3. API app
4. DB (postgreSQL)

**DB (postgreSQL) :** Create a RDS instance for postgreSQL create a database. Once the instance of DB is created it is provided with host, port number and password to create DB, using that credentials we can connect to DB where tables are created with query shared below at the end of this docs

**Create Instance in aws to run app**

Before hosting this applications it's necessary to install nodejs with commands below

sudo apt install nodejs

Sudo apt install npm

Make sure all the necessary node dependencies are installed by checking their version using below commands :

Node -v

Npm --version

**API app :** Host node app with all source files in github with link shared below, before hosting this application please add the credentials of AWS secret key, accessID and DB instance. Which is not uploaded to git for security purpose

<https://github.com/adventum-dev/gwl_adventum_api.git>

Command to run node app in development:

Npm i

npm run dev

Command to run node app in production :

Npm i

npm run build

**Annotation Tool :** with all the source files in github please replace all link with localhost to actual link one gets from hosting of api and point all images to S3 bucket.

Source files of this application is uploaded to GitHub of ID adventum-dev account and same link is shared below

<https://github.com/adventum-dev/gwl_user.git>

Command to run meteor app in development :

Npm i

Meteor

Command to run meteor in production:

Npm i

Meteor build

**Admin App :** This is react app with all the files in github replace links with localhost to actual link got from hosting node app

<https://github.com/adventum-dev/gwl_adventum_admin.git>

Command to run react app in development:

Npm i

npm start

Command to run react app in production:

Npm i

npm run build

**Query to create Database :**

**Total 5 table with query given below**

CREATE TABLE public.folders

(

uuid text COLLATE pg\_catalog."default" NOT NULL,

doctor\_uuid text COLLATE pg\_catalog."default" NOT NULL,

patient\_uuid text COLLATE pg\_catalog."default",

folder\_id character varying COLLATE pg\_catalog."default",

folder\_status character varying COLLATE pg\_catalog."default",

created\_by text COLLATE pg\_catalog."default",

created\_time bigint,

updated\_by text COLLATE pg\_catalog."default",

updated\_time bigint,

last\_labelled character varying COLLATE pg\_catalog."default",

active boolean,

CONSTRAINT folders\_pkey PRIMARY KEY (uuid),

CONSTRAINT folders\_doctor\_uuid\_fkey FOREIGN KEY (doctor\_uuid)

REFERENCES public.users (uuid) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

NOT VALID,

CONSTRAINT folders\_patient\_uuid\_fkey FOREIGN KEY (patient\_uuid)

REFERENCES public.patient\_info (uuid) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

NOT VALID

)

CREATE TABLE public.images

(

uuid text COLLATE pg\_catalog."default" NOT NULL,

user\_uuid text COLLATE pg\_catalog."default" NOT NULL,

folder\_id character varying COLLATE pg\_catalog."default" NOT NULL,

image\_id character varying COLLATE pg\_catalog."default" NOT NULL,

status character varying COLLATE pg\_catalog."default" NOT NULL DEFAULT 'not opened'::character varying,

created\_by text COLLATE pg\_catalog."default",

created\_date bigint,

updated\_by text COLLATE pg\_catalog."default",

updated\_date bigint,

isactive boolean DEFAULT true,

labelled\_images\_id character varying COLLATE pg\_catalog."default",

CONSTRAINT images\_pkey PRIMARY KEY (uuid),

CONSTRAINT image\_uuid\_fk FOREIGN KEY (user\_uuid)

REFERENCES public.users (uuid) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

NOT VALID

)

CREATE TABLE public.patient\_info

(

uuid text COLLATE pg\_catalog."default" NOT NULL,

study\_id character varying COLLATE pg\_catalog."default",

age integer,

gender text COLLATE pg\_catalog."default",

date\_of\_birth character varying COLLATE pg\_catalog."default",

date\_of\_scan character varying COLLATE pg\_catalog."default",

device text COLLATE pg\_catalog."default",

eye text COLLATE pg\_catalog."default",

last\_labelled character varying COLLATE pg\_catalog."default",

created\_time bigint NOT NULL,

created\_by text COLLATE pg\_catalog."default",

updated\_by text COLLATE pg\_catalog."default",

updated\_time bigint,

active boolean,

CONSTRAINT patient\_info\_pkey PRIMARY KEY (uuid)

)

CREATE TABLE public.session

(

uuid character varying COLLATE pg\_catalog."default" NOT NULL,

user\_uuid text COLLATE pg\_catalog."default" NOT NULL,

token character varying COLLATE pg\_catalog."default" NOT NULL,

login\_time bigint,

logout\_time bigint,

active boolean NOT NULL,

labelled\_image integer,

CONSTRAINT session\_details\_pkey PRIMARY KEY (token),

CONSTRAINT session\_uuid\_fk FOREIGN KEY (uuid)

REFERENCES public.users (uuid) MATCH SIMPLE

ON UPDATE NO ACTION

ON DELETE NO ACTION

NOT VALID

)

CREATE TABLE public.users

(

uuid text COLLATE pg\_catalog."default" NOT NULL,

user\_name text COLLATE pg\_catalog."default" NOT NULL,

gender text COLLATE pg\_catalog."default",

phone bigint,

email character varying COLLATE pg\_catalog."default",

password character varying COLLATE pg\_catalog."default" NOT NULL,

created\_by text COLLATE pg\_catalog."default",

created\_time bigint NOT NULL,

updated\_by text COLLATE pg\_catalog."default",

updated\_time bigint,

active boolean DEFAULT true,

user\_type character varying COLLATE pg\_catalog."default" NOT NULL,

name text COLLATE pg\_catalog."default",

CONSTRAINT users\_pkey PRIMARY KEY (uuid)

)