APP DEPENDENCIES

Front-end

```
@angular/common

"@angular/core"
"@angular/forms"
"@angular/http"
"@angular/platform-browser"
"@angular/platform-browser-dynamic"
"@angular/router"
"@ionic-native/core"
"@ionic-native/splash-screen"
"@ionic-native/status-bar"
"@ionic/angular"
"core-js"
"tslib"
"rxjs"
"zone.js"
```

Backend

```
"aws-sdk"

"bcryptjs"

"body-parser"
"cors"
"dotenv"
"email-validator"
"express"
"jsonwebtoken"
"pg"
"reflect-metadata
"sequelize"
```

APP DevDEPENDENCIES

FRONT-END

```
"@angular-devkit/architect
        "@angular-devkit/build-angular
        "@angular-devkit/core"
        "@angular-devkit/schematics"
        "@angular/compiler
        "@angular/compiler-cli"
        "@angular/cli
        "@angular/language-service
        "@types/jasmine
        "@types/jasminewd2
        "@types/node
        "@typescript-eslint/eslint-plugin
        "@typescript-eslint/parser
        "jasmine-core
        "karma
        "jasmine-spec-reporter
        "karma-chrome-launcher
        "karma-coverage-istanbul-reporter
        "karma-jasmine
        "karma-jasmine-html-reporter
        "@ionic/angular-toolkit
        "codelyzer
        "protractor
        "ts-node
        "tslint
        "typescript
```

Backend

```
      @types/bcryptjs"
        "@types/jsonwebtoken
        "@types/bluebird
        "@types/cors"
        "@types/express"
        "@types/node"
```

```
"@types/sequelize"
"@types/validator"
"@typescript-eslint/eslint-plugin"
"@typescript-eslint/parser"
"chai"
"chai-http"
"eslint"
"eslint-config-google"
"mocha"
"ts-node-dev"
"typescript"
"sequelize-typescript"
```

AWS SERVICES

RDS: Postgres database

```
POSTGRES_HOST="udagram.cpqak3wevy9b.us-east-2.rds.amazonaws.com"

DB_PORT=5432

DB_NAME: udagram
```

S3: IS UDAGRAM FRONTEND POINT

URL: http://udagram-957284979758-bucket.s3-website.us-east-2.amazonaws.com

PLZ: GO TO THIS LINK TO TEST API FUNCTIONULITY

Elastic Beanstalk:

URL: http://udagram-env.eba-nt44bg5f.us-east-2.elasticbeanstalk.com/

PROJECT WALK THROUGH

- Clone my project from this link: https://github.com/is-is/udagram-hosting.git
- Install frontend and backend packages exist in package.json files by running npm install in both udagram-frontend and udagram-api
- Build and Deploy for backend is done by running npm run build then by creating a file [deploy.sh] to initiate then create elastic beanstalk environment --- setting

up environment variables exist on .env via circleci which will set the ones exists on elastic beanstalk

- For frontend run: npm run build then npm run deploy to upload app on AWS s3 web hosting bucket created via AWS cloud services [I have created a s3 bucket, made it public and enabled static web hosting]

- Create RDS on AWA to store client's data, connect it with the udagram app by referring to it on DB configuration of our app

Finally deploy our elastic beanstalk environment in our app called udagram-env then test by URL: http://udagram-957284979758-bucket.s3-website.us-east-2.amazonaws.com

ALL PREVIOUS STEPS DONE AUTOMATICLLY ON CIRCLECI PIPELINE

First we sign into circleci via github account

set up environment variables

create .circleci/config.yml in the root of our app

in config.yml circleci does the following jobs

Build

1. Set up environment
2. Prepare environment variables
3. Install Node.js
4. Checkout code on github for changes
5. Install front-end dependencies
6. Install backend dependencies
7. Build frontend
8. Build backend

Deploy

1. Setting up Elastic Beanstalk CLI
2. Install AWS CLI
3. Configure AWS Access Key ID
4. Checkout code

5. Deploy frontend and backend