Catalog

Individual-Health-Record Team Awesome

Fall 2018 CS6440

Table of Contents

Cover Page	1
Project Structure	2
Elements	4
ihr-webapp	4
Reasearch	6
Document	6
.dockerignore	6
docker-compose.yml	6
ihr-webapp.sln	6
ihr-webappdb-docker	
ihr-webapp-docker	
Jenkinsfile	
package-lock	7
README	7

Cover Page

Team Awesome

IHR

Individual Health Record

Github Link

https://github.gatech.edu/gt-cs6440-hit-fall 2018/Individual-Health-Record-IHR-Prototype

Table 1. Team Members

Name	Role
Joseph Honea	Developer, QA
Eric I-Hsin	QA, Developer
Dionisios Gizas	Developer, PM
Emma Yuhan Zhang	PM, Analyst, QA

Project Structure

- ihr-webapp
 - node_modules
 - public
 - images
 - js
 - chartCreator.js
 - listCreator.js
 - routes
 - dashboard.js
 - history.js
 - login.js
 - logout.js
 - tester.js
 - UserDataEntry.js
 - utils
 - dbutil.js
 - fhirUtil.js
 - views
 - dashboard.pug
 - history.pug
 - index.pug
 - login.pug
 - UserDataEntry.pug
 - app.js
 - config.js
 - ihr-webapp.njsproj
 - 。 package JSON
 - package-lock JSON
 - README.txt
- Reasearch
- Document
- .dockerignore
- docker-compose.yml

- ihr-webapp.sln
- ihr-webappdb-docker
- ihr-webapp-docker
- Jenkinsfile
- package-lock
- README

```
File Edit Selection Yiew Go Debug Terminal Help
                                                                                                                                                                                                                                                                                            JS dbutil.js ×
                                                                                                                                                                                                                                                                                                                                                                                                                                                            <u>®</u> ■
 a
                                                                                                                                    var mongoose = require('mongoose');
var schema = mongoose.Schema;
var fhirutil = require('./fhirUtil');
               ▲ OPEN EDITORS 1 UNSAVED
                   # main.css public\stylesh
• @ dashboard.pug views
  Q
                                                                                                                                      var userSchema = new schema({
   oid: { type: String, required: true, unique: true }, //Provided by Azure AD
   name: String, // Name of user from Microsoft Account
   email: String, // Email From Microsoft Account
   isProvider: Boolean, // Is this user a provider
8
r patientSchema = new schema({
    userInfo: userSchema,
    fhirResources: [{
        unl: String, // URL for fhir resource
        patientId: String, // Patient Id from resource
        validated: Boolean // Was this user Validated for this resource?
}], // Array of fhir resources for user
medications: {{
        resourceUrl: String, // fhir resource url, or MANUAL for manual entries or PROVIDER for provider entries
        resourceId: String, // Resource Id from fhir resource. null if manual or provider
        name: String, // Name of medication
        instructions: String,
        frequency: Number, // How many times per day should it be taken
        period: Number,
        periodUnit: String,
        doseQuantity: Number, // How much should be taken per dosage
                                                                                                                                     var patientSchema = new schema({
                    JS fhirUtil.js
                   ≣ ihr-webapp.njsproj
                  {} package-lock.json

    package.json
    README.md

                                                                                                                                                         doseQuantity: Number, // How much should be taken per dosage dosellnits: String // Units for dosage
                                                                                                                                                                                                                                                                                                                                                                                                                           ▼ ≝ 🖆 🗅 ^ ×
                                                                                                                                                                                                                                                                                                                                                                              Git
                                                                                                                      > git for-each-ref --format %(refname) %(objectname) --sort -committerdate
                                                                                                                      > git remote --verbose
                                                                                                                       > git status -z -u
> git symbolic-ref --short HEAD
                                                                                                                      > git rev-parse master
> git rev-parse --symbolic-full-name master@{u}
> git rev-list --left-right master...refs/remotes/origin/master
> git fon-each-ref --format %(refname) %(objectname) --sort -committerdate
                                                                                                                     > git for-each-ref --format %(refname) %(objectname) --sort -com
> git remote --verbose
> git status -2 -u
> git symbolic-ref --short HEAD
> git rey-parse master
> git rev-parse --symbolic-full-name master@{u}
> git rev-list --left-right master...refs/remotes/origin/master
OUTLINE
                                                                                                                                                                                                                                                                                                                                                 Ln 1, Col 1 Spaces: 4 UTF-8 with BOM CRLF JavaScript 😃 🔔 1
```

Figure 1. Project Structure

Elements

ihr-webapp

node_modules

Local install puts package and dependency in this folder

public

images

folder for images used in website

js

chartCreator.js

javascript for create chart to display user health information

listCreator.js

javascript for create list to display user health information

routes

folder for javascript files used for each webpage

dashboard.js

javascript for dashboard webpage

history.js

javascript for history webpage

login.js

javascript for login webpage

logout.js

javascript for logout webpage

tester.js

javascript for tester webpage

UserDataEntry.js

javascript for user data entry webpage

utils

folder for javascript contains database utility

dbutil.js

javascript for database webpage

fhirUtil.js

javascript for FHIR server webpage

views

folder for pug files

dashboard.pug

pug file for dashboard page

history.pug

pug file for history page

index.pug

pug file for index page

login.pug

pug file for login page

UserDataEntry.pug

pug file for UserDataEntry page

app.js

javascript to run the app

config.js

javascript for some configuration

ihr-webapp.njsproj

njsproj file that run VS up with additional help that's specific to Node.js

package JSON

JSON file for package

package-lock JSON

JSON file for package dependency

README.txt

readme file automatically generated

Reasearch

folder that keep all the past research done for the project

Document

folder that keep all the adoc files and pdf based on adoc files

.dockerignore

file to list files and directories to exclude

docker-compose.yml

config file for docker-compose.

It allows to deploy, combine and configure multiple docker-container at the same time

ihr-webapp.sln

structure file used for organizing projects in Microsoft Visual Studio

ihr-webappdb-docker

docker file for database

ihr-webapp-docker

docker file

Jenkinsfile

File that helps to automate the non-human part of the software development process, with continuous integration and facilitating technical aspects of continuous delivery

package-lock

lock version

README

readme file automatically generated <<<