Sets password

* The app allows users to create an account,
* Log into the account
* Sign back out securely.
* All user data is stored in a mysql database.

User story

When someone who wants to safely log in to the database or site They want to know their personal details are safely stored

* SEQUELIZE
* LOCAL
* MYSQL2
* EXPRESS
* EXPRESSSESSION
* BCRYPTJS
* PASSPORT
* PASSPORT

Getting going

* Clone repository to terminal location
* Once this is complete
* Follow these steps;
* Create a mysql db called "passport\_demo"
* Config file
* open config.js
* add your data username, password etc
* Open terminal in current repo
* run "npm install or npm i" to install all node packages
* "node server.js"
* connect to server
* Open browser
* "[http://localhost:8080](http://localhost:8080/)" in search bar

FILES

Config

* config.json
* connection configuration to connect to server
* passport.js
* contains javascript logic that tells passport
* log in with an email address and password

Middleware

* isAuthenticated.js
* restricts routes
* if user is logged in
* continues with request
* config.json configuration to connect to server
* passport.js
* log in with an email address and password

Models

* index.js
* connects to db
* imports log in data
* user.js
* requires "bcrypt" password hashing
* database secure
* JS that tells you what is stored on the database
* ROUTES
* apiroutes.js
* routes for signing in
* logging out
* users specific data to be displayed
* htmlroutes.js
* routes makes sure user is signed in
* If user has account
* sends them to html page
* package.json
* contains all package info
* node modules used
* version
* server.js
* requires packages
* sets up PORT
* creates express
* middleware
* creates routes
* syncs database
* logs message in terminal

Link to google docs

https://docs.google.com/document/d/1-mk5SZe\_m8PCLTTwScqwjRqs0hay5eu1fA2rBwfBMCI/edit?usp=sharing