**CAPSTONE PRESENTATION**

SAESHA

This application supports the functioning of a business that serves the real estate industry. By capturing high-quality 360-degree images, Sphyn assists real estate agents and homeowners with demonstrating a home’s appeal to potential buyers. On the back end, the Sphyn website utilizes Express to run the server and routes and Sequelize to create and maintain database tables, all of which is facilitated by Node.js. On the front end, React splits functionality between different components, which all serve a particular role.

HOME PAGE

The user’s experience of the site starts with the home page. This page displays information about available service packages, along with links that allow a new user to register and a continuing user to log in.

HARLEY

ACCOUNT

An account is created by entering a username, email, password, and phone number into a form. Customer data is saved to the database. Passwords are protected with bcrypt hash encryption so that any database breach will not compromise user accounts.

LOGIN

After creating an account, the user is sent to the long page. If a correct email and username is entered, most users will be routed to their profile page.

YING

ADMINISTRATION

Administrators are identified by email address during the login process. The administration page allows for editing of anything in the database’s Services table, including adding an image URL to the database. This addition allows customers to access the completed 360-degree image for that particular order. Payment status (or any other field) is also updated using this form.

BOOKING SERVICES AND PAYMENT

The booking component consists of a form that allows a customer to request a service – that is, for photos to be taken of a particular property. The form requires a nickname, address, city, state, zip code, and package selection. Upon submission of this form, the PayPal payment screen displays. Upon selecting a payment option, the user is routed to the PayPal site, where the payment is made. After payment, the user is routed back to their profile page, which lists the requested service.

NICK

PROFILE PAGE AND SHOWCASE

The profile page greets the customer and displays any pending or completed orders. Login is required for access to this page. To allow for public access to photos, the showcase page displays all services that have been requested by a particular customer. This customer is identified by entering the corresponding email address into a form. When scanned, the QR code on both pages leads to the showcase, allowing for sharing with mobile devices.

GOOGLE MAPS

The Google Maps API is used on the showcase page. It displays the location of a home by accessing the address entered by the customer and determining the address’s latitude and longitude.

ERROR REPORTING

A user not logged in will receive an error message stating “User not authorized. Please log in” when trying to access the profile page. The words “log in” function as a link to the login page. An unauthorized logged-in user attempting to access the administration page will see the error “Unauthorized user. Please return to your profile page,” with the words “profile page” functioning as a link to that user’s profile page. A user who is not logged in who tries to access the administration page is routed to the login page. A not logged-in user trying to submit the booking form is routed to the login page.

PRINCETON

POTENTIAL UPDATES

The following features would enhance this application:

* On the home page, a user can click on a service and have that service selected in the booking form
* Update payment status automatically with PayPal data

DATABASE & SITE MAPS

(Briefly explain these in the README.)

YING

* Massive merge conflict with github
* Google Maps - passing the lat and lng into google maps
* Understanding differences between how the express api vs client routes works
* definitely hooks, it's a been a learning process