PROJECT 2: PLAYWRITING WEB SITE

I am going to make my personal web site for my Playwriting work. It will include:

* A landing page with an animation with my name and “playwright” as well as an “Enter here”.
* An “about” page.
  + This is the opening page after the animation. The upper half is a carousel of production photos.
  + Over-laying the middle of the carousel is my headshot.
  + The main material is a simple ‘About’ statement.
* An index page
  + It lists all of my full-length and short plays with links to their show page
  + Limited info: Title. Logline. Comedy/Drama.
* A show page
  + Each play links to the show page. These include a large production photo, synopsis, actor requirements, development history, production history, awards, quotes from the play.
* A new play form
  + Only accessible by Admin. It includes spots to fill in all the information that will be required for the show and index pages.
* An Admin-Only index page which allows me to update each play.
  + May be able to turn this into some kind of hidden nav that is only revealed when the user is logged in. It simply adds the buttons when logged in.
* A contact form – STRETCH
  + A form for people to send me an e-mail.
* Collaborators – STRETCH
  + A page with Theatres/Orgs/Artists I’ve worked with in the past. Maybe have a series of photos from some of these experiences. If the user clicks on the org, a series of photos is pulled from a deck, when they click on a different one, those photos return and a different series is pulled out.
* A password sign-in in order to access the new play creation page. FOOTER
* A sticky nav that has a spot for “About”, “Play Overview”, “Individual Plays”
* A sticky footer with: copyright info, “Contact”, “Admin login”

Other Possible Additions

* A ‘News’ feature
* Links to Organizations you’ve worked for and Artists you want to highlight.

Talking With John RE: Project 2

* Would be better to build it with React.
* You could have a separate model for Users with you as the only user. That way you can gain experience with multiple models.

SCHEDULE

Monday 7/8:

* ~~Brainstorm site~~
* ~~Work on some wireframes, explore software~~
* ~~Wireframing – Explore the Adobe Design software that you have.~~
* Watch Bootstrap, ~~Materialize~~, Skeleton tutorials

Tuesday 7/7:

* ~~Heroku tutorial 9:00~~
* Gather your images into one folder – Make sure they are good quality
* ~~Project Approval 11:00~~
* ~~Organize how your data is stored and how it will navigate~~
* ~~Build the New Page/ Server.js /Controller/Index Page~~
* ~~Build the Model Schema for Play : IMAGE, TITLE, DIALOGUE SAMPLE, CAST NEEDS, COMEDY/DRAMA, SYNOPSIS, DEVELOPMENT HISTORY, HONORS~~

Wednesday 7/8

* ~~Stand-Up Check-In 11:00~~
* ~~Add User Log-in and Password – look through notes~~
* ~~Finish the RESTFUL CRUD steps: Edit, Update, Delete~~
* ~~Plot out design – shared attributes between pages – thinking about the libraries and classes that you’ll use~~

Thursday 7/9:

* ~~Stand-Up Check-In 11:00~~
* ~~Check out Heroku again and see if you still get the error. If so, look at the troubleshooting guide and then talk to someone.~~
* ~~labels in your new user ejs file form.~~
* ~~New Play form: Add genre, add length checkboxes, add a “story” textarea.~~
* ~~Show Page: Add genre, add length checkboxes, add a “story” area.~~
* ~~Add all the new information to your EDIT form~~
* ~~Make the Casting stuff actually display on the page.~~
* ~~Write the logic for your new checkboxes. If they are checked, they will display certain text (ie comedy, drama)~~
* ~~Make it so Roles disappears if it’s zero.~~
* ~~Create a seed page to start the design process.~~
* ~~ILP WORK 4:00~~

Friday 7/10:

* ~~Stand-Up Check-In 11:00~~
* ~~Design: Nav bar (partial), footer (partial).~~
* ~~Get your form buttons to add input fields as you press “Add-One” (require text in the field in order for it to add to the array.)~~
* ~~Get Array to work~~

Saturday 7/11:

* ~~Require password to Edit, Update, Create New, and Delete~~
* Watch Bootstrap, ~~Materialize~~, Skeleton tutorials
* ~~TA time 10:00~~
* ~~If you are logged in, the EDIT and DELETE buttons will appear for editing. The Index page shouldn’t need anything added to it.~~
* ~~Get the image middleware working~~
* ~~Get all those Dev Opps and Honors fields to go into the Object’s array.~~

Sunday 7/12:

* Overflow day
* STRETCH: Collaborators Page
* STRETCH: Contact Page – how does a message get forwarded? Or can it all get logged in a db somewhere?
* STRETCH: News section (could maybe be a part of the About page. Maybe with it’s own update form).
* Stretch: Set up the index page so it will separate the play data and put them under the correct length headings
* Design: Do a basic blocking out of show, index, and about pages w/ Lorem Ipsum for now. Don’t get caught up in the details.

Monday 7/13:

* Overflow day
* Design: User Creation form, Log-In form, add play, edit play. Keep these simple
* STRETCH: Figure out proper error calls and routing
* STRETCH: Make casting so it can’t go below zero. If it does, an error message should pop up.
* Stand-Up Check-In 11:00
* README document

To Do

* Ask Travis how he dealt with his multiple buttons
* STRETCH If an input box in an array element is empty, don’t push that empty string into the array. Make it so you can delete items from the array (for example if you want to edit something out of the current array.)
* STRETCH Have a form section for uploading multiple production photos to the homepage carousel. (This is on top of having one for uploading a single production photo for your background display). You will need to go back to the stackabuse.com article and set it up for multer().array()
* STRETCH Get the Update form to update image.
* Get the remove file function working so you can take off image files in the edit form.
* Set up your users/new route so, you can’t see that set-up in the docs. Hide it in your .env file.

NOTES:

* Focus on MVP!!
* Schema default can be a function.
* It seems that I keep getting logged out of the site. This is a new development. (Since I hid the New User form?)

TA Time

* How to work that empty array in show.ejs more cleanly.
* Is there a way to get rid of all of these if statements in show.ejs
* The radio buttons all go unmarked when I go to edit. Can I keep the marked one marked?
* Adding images to an object through a form – for the update route
* How to see what’s in the Heroku database?

CAROUSEL

* You can limit the width of your slide carousel if you’re worried about how only showing 2 at a time will effect the layout on a large screen.

Question:

Add alt info to images and form questions

What’s the deal with using “fruits.push(req.body)” in the post function?

How do I know where to place things in the server.js file. Asking because my app.use –session stuff was lower down and so it wasn’t working.

Can a Number datatype in your Schema allow a range of numbers? – You may need to do a “Min. field” and a “Max field”

DESIGN NOTES

NAVBARS

* Bootstrap has notes on making a navbar collapse into a hamburger menu when the screen gets small. It’s in the Navbar section of getbootstrap.com
* Use .active to make it clear that the user is currently on that page – LOOK INTO THIS TO SEE HOW IT ACTUALLY WORKS.
* You can add something to a nav item to make it deactivate if it is the current page.
* You should designate strings of text when in the bar, so it knows how to treat them. Check out ‘Text’ in the link above.
* The order of your nesting IRT <nav> /<div container> can effect the doubling up of padding. You’ll want to avoid this doubling up when dealing with smaller screens especially. See: ‘Containers’ under the Navbar section of the docs.
* I think you want want fixed-top for the navbar.

ISSUES

STAND-UPS

I learned today that radio buttons need to have String values in the Schema

I learned today that empty arrays still have a value.

Question

CHANGES MADE THAT MIGHT CAUSE PROBLEMS

-Check into the form action if file upload isn’t working correctly.

TROUBLESHOOTING

* Middleware ALWAYS needs to be above the controllers
* Check the browser address bar and see what’s there