



Fullstack Engineering

**ADAPT FASTER TO THE
FUTURE OF WORK!**

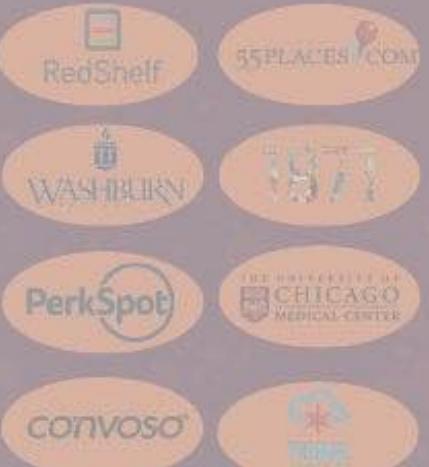
WE PROVIDE ARTISTS AND
PROFESSIONALS FOR ANY CORPORATE
EVENT. WE DO IT ALL.
BOOK SPECIFIC ARTISTS OR SEND US A

REQUEST AT:

WWW.QOOLEY.COM



OUR CUSTOMERS:



"QOOLEY REALLY PUT THE TIME AND EFFORT INTO HELPING US GET A GREAT COMBINATION OF ARTISTS. THEY WERE ENGAGED, INFORMATIVE, HELPFUL, AND TOTALLY INVESTED IN THE SUCCESS OF OUR EVENT. WOULD NOT HESITATE TO RECOMMEND!"

Sara
PerkSpot

Qooley

CHALLENGE

SOLUTION

RESULTS

OUR MISSION

Our mission is to bring joy, wellness, & productivity at your organization.

Here are our core beliefs:

- Connected and Engaged employees understand their purpose and mission.
- Happier employees are healthier and perform better at work.
- Gamification of training and onboarding activities leads to increased productivity.
- Making sales and marketing calls more fun helps win over customers.

We want to help you leverage our creative solutions to stay ahead of the trend.

EVENT EXAMPLES

- Virtual happy hours
- Team-building events
- Team Celebrations & Parties
- Onboarding & Trainings
- Conferences & Fairs
- Awards & Recognition events

PRICING

WE MATCH YOUR BUDGET - SEE CATEGORY RANGES BELOW



COMEDY

<\$100
\$100-\$500
\$500-\$1000
>\$1000

IMPROV

<\$100
\$100-\$500
\$500-\$1000

MAGIC

\$100-\$500
\$500-\$1000
>\$1000

CLASSES

<\$100
\$100-\$500
\$500-\$1000
>\$1000

CONTACT US



@QOOLEY_OFICIAL



@QOOLEY_OFICIAL



@QOOLEY_OFICIAL



@QOOLEY_OFICIAL

My Fullstack Engineering Experience Begins With...

Jake, Dean, and Bobby's Car Site

Qooley.com.

Restaurant Reviews at Carleton College

The Online Virtual Learning Site, Chalkboard.

Data Breach Notification Laws-R Shiny App.

Tech Companies, Culture, & Local Tidbits! Qooley Frontend-Test.

How Do Pain Devices Work? Celéri Health.

My First Fullstack Academy Project

Stackathon 1. Graphing .csv files and Data Visualization

Carleton Case Studies in Data Science

MLH - Welcome to Our Pod Website

Olaf's Swim

Visual Studio Code GitHub Projects.

Fun Facts Website: Request Your Own Fun Facts.

Qooley App For Company Parties

QooleyBiz App for Artists

Welcome to My Blog

Internet Radio Streaming

Coffee in CSS3

Chess Site

Wolfram Summer School 2022-Adventures in Multi-way Computation.

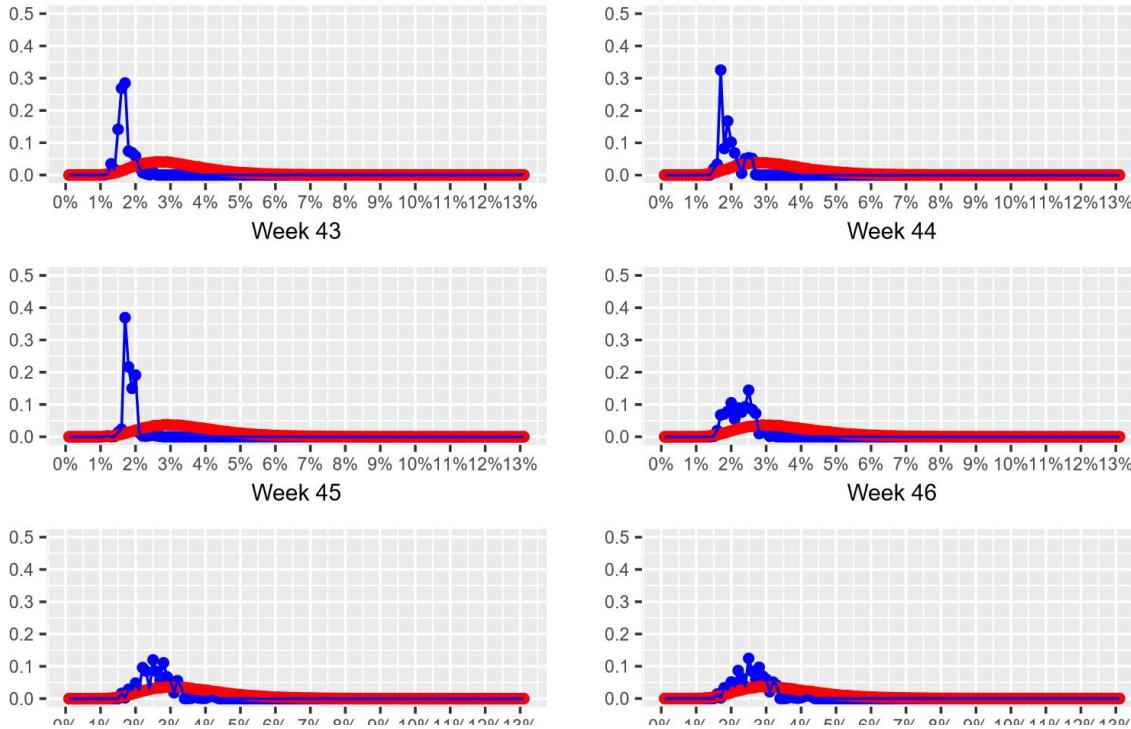
Guessing Game.

The Mangrove.

My Fullstack Engineering Experience Begins With...

My First Internship-Biostatistics at The Reich Lab

By Dean Gladish and Professor Nick Reich of the University of Massachusetts



Among hosting interns from Carleton College, the Reich Lab is primarily interested in predicting influenza outbreaks with a focus on comparisons between models submitted to the CDC's Flusight challenge. Our EM algorithm, SARIMA, KS-test, Chi-squared test, have all been phenomenal. But what's more is the fact that we wrote a [final report](#) and were able to contribute to the lab.

The programming language applied is R; it appeals to the coding side of me and is efficient in terms of executing our type of predictive power. This exemplifies not only predictive power but also the principles of sorting data, chronologically sorted, the chance that isn't usually mentioned that is to employ the Data Structures in Statistical Programming and Linear Regression. All plots are made in ggplot2.

Jake, Dean, and Bobby's Car Site

By Dean Gladish, Jake Armijo, and Bobby Ton



The best of all is being a participant in a car sales website. We have not only deployed it with the admin account at username: admin@email.com, password: txtz7pap3FmoHxn, and [our Git repository](#) is amazing.

My taste in making a car website, whether it's designing search filters, tangling with image URLs, or querying databases with Sequelize, my site exemplifies the team-oriented spirit of PostgreSQL-model Object-Relational Mappers, for which Jake & the team drew inspiration when Bobby implemented the Sequelize models. Working together with him on Socket I/O and learning about how data is generated via Faker.js, is a reminder of how important it is for me to work in versatile teams. My teammate Jake for example understands the nature of post-builds and Babel; on the conceptual framework he excelled. I couldn't have coded it better myself.

If you're fascinated by our site, [then here's the link to access our homepage](#).

Qooley.com.

[Look here for company parties and Qooley company kick-off events!](#)

Qooley

Boost engagement and retention at your company!

A man in a tuxedo and bowler hat performing a magic trick.

Artists & Entertainers

A person wearing a VR headset looking at a landscape.

Immersive Experiences

A speaker giving a presentation to an audience.

Speakers & Coaches

People sitting at a table in a restaurant.

Venues

A basket filled with various items like hats and bags.

Company Swag

A decorated event space with tables and hanging lights.

Event Services

What We Specialize In

ONBOARDING

TEAM BUILDING

ENGAGEMENT

PERKS & REWARDS

People working together at a desk.

A group of people in a circular formation, possibly playing a game.

A group of people taking a selfie together.

A tropical beach scene with palm trees and a clear blue sky.

Clients & Testimonials

"Qooley really put the time and effort into helping us get a great combination of artists. They were engaged, informative, helpful, and totally invested in the success of our event. Would not hesitate to recommend!"

-Sara, PerkSpot

CONVOSO

Contact Us

www.qooley.com admin@qooley.com @qooley_official QooleyOfficial [@qooley_official](https://Qooley)

The one and only gooley.com is the website that I built myself. We did a lot of team meetings, let's dive in!

On the [About Us](#) page, you can see that we started out in June of last year. When we did the meet and greet, I met a student of byzantine complexity and his name was Sohan Sarabu. He handed this site to me and I'd like to give him a special shout out for introducing the Next.js routing system. That is how we easily display everything in the ./pages folder.

If you're interested in how we got started, there's so much to chat about. Whether it's our Database Storage Buckets and the artists & experts signing up in there, multiple image development... how we keep the site running, it's all really up to the artists. Moreso than that we did the four-stage testing paradigm, local to two testing servers to the production server. It's amazing. We even wrote a [documentary](#) about it.

<https://www.gooley.com/> is our home page.

Restaurant Reviews at Carleton College

Mississauga Restaurant Search!

Perlman Database Workshop.



[Restaurant Reviews Website](#) was built by three students - me, Hashir, and Tony. We did it to connect the Epicurean spirit with specific restaurants, and discovered many different use cases. It's a whole new level and established way of working with traditional SQL and build a [Python website](#). On the foundation of HTML5, CSS3, and SQL, we set the origin of our site to be derived from an SSH into the Perlman database at Carleton College, connecting the Epicureans among us.

While we were building the site, we talked about some human rights issues; when we were just starting to learn the Git Version Control system (which takes up a lot of bytes, doesn't it?), we learned about how the ICE contract with Git was violating human rights. It was some of the most talkative, compassionate, and enthusiastic classes that I've ever had. We got all kinds of user constituencies like Restaurant Customers, Food Critics, those are our constituents.

The Online Virtual Learning Site, Chalkboard.

Khuong Le, Dean Song Gladish, Zach Bryce, & Jonathan Arreola

The screenshot shows a dark-themed web application titled "Chalkboard". At the top, there's a navigation bar with links for "Home", "Teacher Dashboard", and "Logout". Below the navigation, there's a table-like structure displaying various class details. The columns include "Information", "Econ", "Senior", "REACTO", "Astrobiology", "Daytime", "Wasting", "California", and "Physic". Each row represents a class, such as "Econ 201" or "Astrobiology". Under "Information", there are sections for "Schedule" and "Time". Under "Daytime", there are sections for "Serials", "Family & Social Roles", and "Internet". Under "Wasting", there are sections for "Time on the Internet" and "Schedule". Under "California", there are sections for "Here we Come" and "Schedule". Under "Physic", there are sections for "Star Trek" and "Schedule". At the bottom of the page, there's a green button labeled "List of Classes" and three rectangular boxes containing text: "Elvish, the language of LOTR", "Information Radio", and "Econ 201".

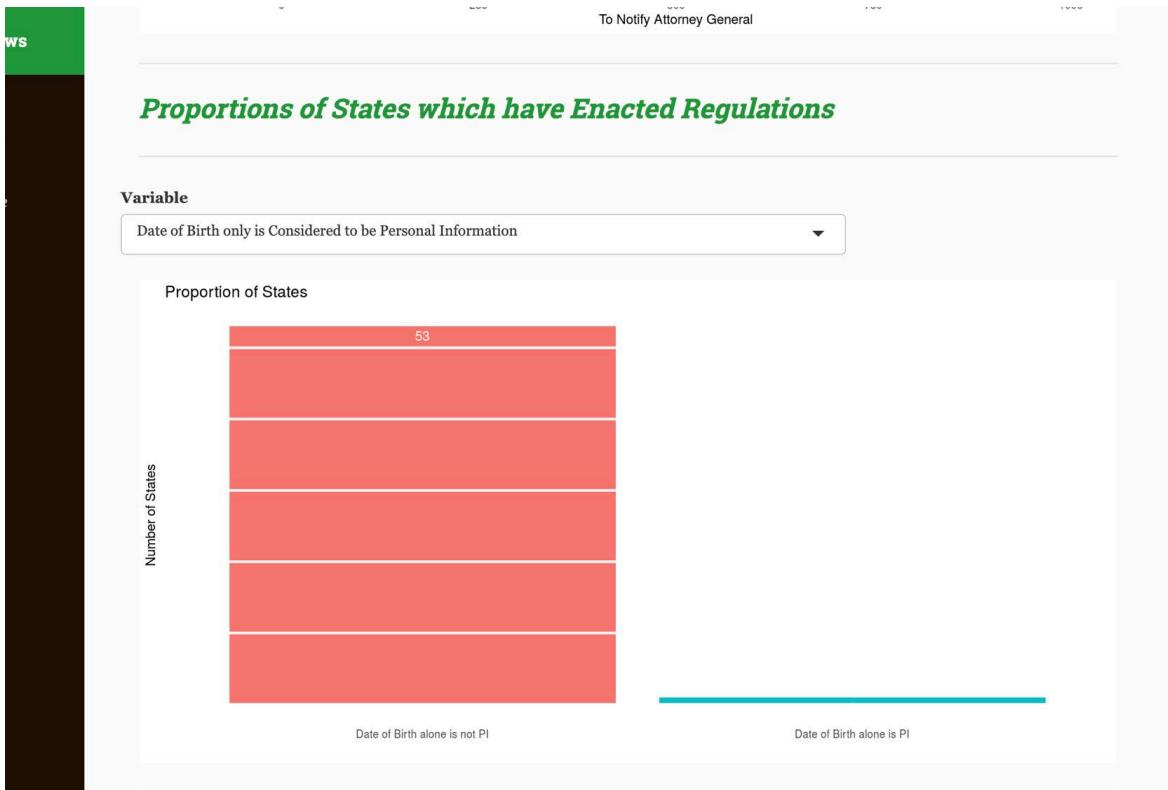
When you're going from Point A to Point B across campus there's a site you can use..or so it seems.

Check out our [Video Explanation](#), it's about what we did to code the site.

We started out deploying on Heroku and then learned how to manage .gitignore, secrets.js, zoomSecrets.js, and most of all how to write postbuild in package.json, and make sure that the packages agree with each other (and how to write start and build commands for servers like Herokuapp, Vercel.app, and [render.com deployment](#)). It really speaks to our skill in re-deployment. We have so many classes on there. When you want, here're some of the logins: email: teacher1@email.com, password: 123 and for the student users email: studentAllClasses@email.com, password: 123 or email: student3@email.com, password: 123.

If you want to engage in live, real-time chat we have built a dashboard for both students & teachers to make new classes and in the student dashboard, view their descriptions, register for & post new classes. We used Excalidraw to make the wireframes that allowed us to understand the roadmap and the relational databases, hash table function generators, (non-)binary tree searches & node traversal algorithm run-time that made this website possible.

Data Breach Notification Laws-R Shiny App.



When I was a Legal Intern at CDK, Global, the company which provides the software powering most of the car dealerships around the Chicago area, we made a lot of bar charts and made significant advances in notification laws for data breach [research](#). I was surprised to see that date of birth is not considered to be personal information.

I sat in on mergers & acquisitions, focused on the elements of Non-Disclosure Agreements, and learned all about Securities Law. With a corresponding Excel Reference Table, I demonstrate the notification laws for data breaches for all 52 states, and tried to include all the shiniest elements of the Integrated Development Environment, R Studio.

The timeless principles of graph theory demonstrate my ability to work with & display tables from Microsoft Excel to the R Shiny programming language. It was an excellent choice and adventure in Data Science.

Tech Companies, Culture, & Local Tidbits! Qooley Frontend-Test.

We've got a secret ./venues page and test server.

Inspiration and Art at My House

Phone Number

2243064486

Street Address

City

3834 Lizette Lane

Chicago, Illinois

Description

People are naturally inspired, they're enthralled by art. Functionality is a given; if people can't get from point A to point B, then nothing else follows. Drawing inspiration from Shopify, HotelTonight, join us in creating an events & crafts website with any colors and designs that come to your mind!

Amenities & Accessibility

When you sign up, you'll get a coupon to let you learn how to play music at Qooley.com. We can recommend perks for employees, develop a community of amazing experts and artists with a lot of talent. We've got expert instructors, teachers and coaches.

What makes us unique?

Travelers can connect with knowledgeable locals who are just energetic, positive, fun people who would want to host an authentic meeting!

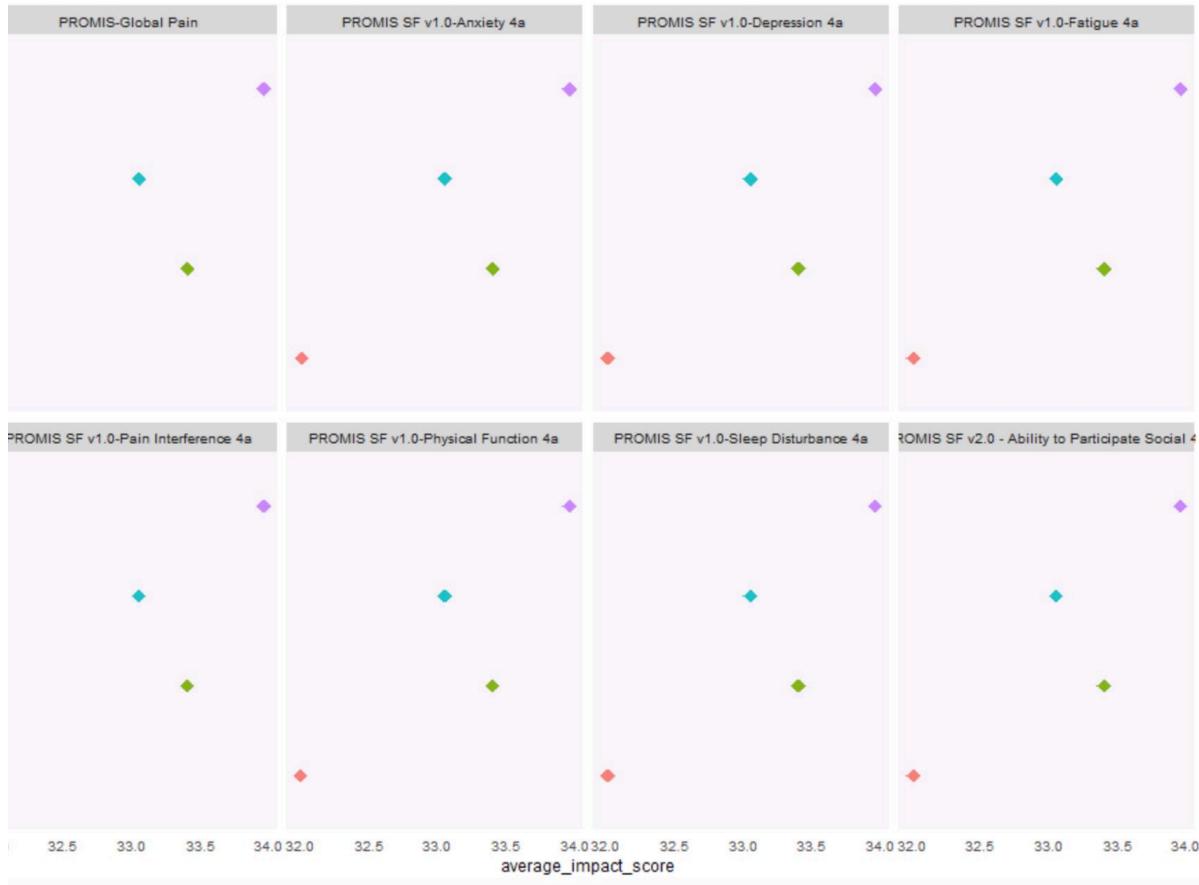
At Qooley.com, we've got a lot of companies to market to; let's include this site twice because it's so fascinating; companies are going through a lot of problems with people leaving their jobs because of the pandemic. Like the people at Major League Hacking, they want more flexibility, and don't want to be stressed out too much. We provide the balance & work-life integration. Our locals are energetic, fun people who could provide an amazing experience because they are knowledgeable and know the secrets of their surrounding environment.

At our kick-off meetings, we strive to emotionally connect; we understand and feel your pain points, when you tell them to us.

Visit us at our official server, qooley.com!

How Do Pain Devices Work? Celéri Health.

Authored by Christopher Hanes & Dean Gladish



At Celéri Health, we analyze the efficacy of pain devices before and after they are implemented. Via real-time [.Rmd script](#) we help pain clinics identify subjective, patient survey data that is self-reported and turn it into [real insight](#) for the patients; based on Health Utilities Index Mark 3 scores and PROMIS surveys, we provide the public with confidence in the data science that is being done.

All programming is done in R; we're doing [Exploratory Data Analysis](#), [Permutation distributions](#), [Chi-squared tests of independence](#), [T-scores...](#) because R allows us to [display interest in statistical significance](#). Dplyr for data manipulation, ggformula graphics functions, tidyverse for mutate-filter-select-slice piping, which shows [comprehensive research](#) into data science. [What we've created](#) is worthy of a podcast of its own.

My First Fullstack Academy Project

[Campuses](#)[Students](#)[Add a Campus](#)[Add a Student](#)

Wake Forest University



Address: 1834 Wake Forest Rd,
Winston-Salem, NC 27109

Description: Wake Forest University is a private research university in Winston-Salem, North Carolina. Founded in 1834, the university received its name from its original location in Wake Forest, north of Raleigh, North Carolina.

Students:

Name: *Indiana Frank*
Email: indiana.frank@mail.com
GPA: 4



Role-playing as the CTO of Margaret Hamilton's Interplanetary Academy of JavaScript has [never been better](#). With regard to RESTful web platforms, you'll find one here in Gentle Hamlet with Express to handle HTTP requests and Sequelize to interface with the database; components that display student and campus data are connected via Redux to the database, and side effects like AJAX requests are encapsulated in thunks.

The form data & field validation, Add a Campus or Add a Student, are managed by stateful React components, while our use of Axios is exemplary in making calls to the backend. Not only this but npm thoughtfully includes the chalk library for coloring the command line. Look at [the video](#).

Stackathon 1. Graphing .csv files and Data Visualization

The screenshot shows a web-based application interface. On the left, there is a list of messages from various users (Syndi, Sherwood, Dean S. Gladish, and several Guests) with their corresponding avatars. A message from 'Sherwood' contains a link to a CSV file. Below the messages is a section titled 'Upload Image' with a placeholder image and a 'Chat!' button.



There's nothing better than [a refreshing data visualization .csv](#) grapher, which is the inspiration for programming a data visualization site. Not only that but as you can see, this site is all mine so feel free to send messages in the chat, given the promise that it's re-deployable. The interesting thing about this one is [the video](#) which, for lack of time, talks all about the chat and data visualization but doesn't mention the logistics like Travis CI/CD in YAML or Prettier instead of ESLint, both are things which you don't see, part of the appeal.

It's clear that this takes after [the Chart.js demo](#). Not only can you upload images and by the way, this is a test of various things like image upload and running the seed.js script from package.json, but also the loadFile helper function which sets the image.src on the DOM elementById output with URL.createObjectURL (event.target.files[0]). Also I've been inspired by Bobby and Jake's illuminating work on JDB cars (which taught me how to use faker.js to generate names for chat and how to color the command line through chalk), and my own stackathon project which is on Chart.js as seen on my blog on page 20.

Carleton Case Studies in Data Science

[Outlier Particle Emissions from Many Cars at An Intersection with Linear Regression Model](#)

[Linear Regression on Mortality and The Logarithm of Air Pollutants](#)

[Binary Accuracy Study Into The Effects of Union, Gender, Race, Age, and Income](#)

5 (The Reich Lab) KCDE and CUBMA

The Reich Lab) KCDE and KDE

The Reich Lab) KCDE and SARIMA
Seasonal Difference = True)

The Reich Lab) SARIMA and KDE
Seasonal Difference = True)

The Reich Lab) SARIMA(Seasonal Difference) and SARIMA(No Seasonal Difference)

The Reich Lab) KCDE and CUBMA

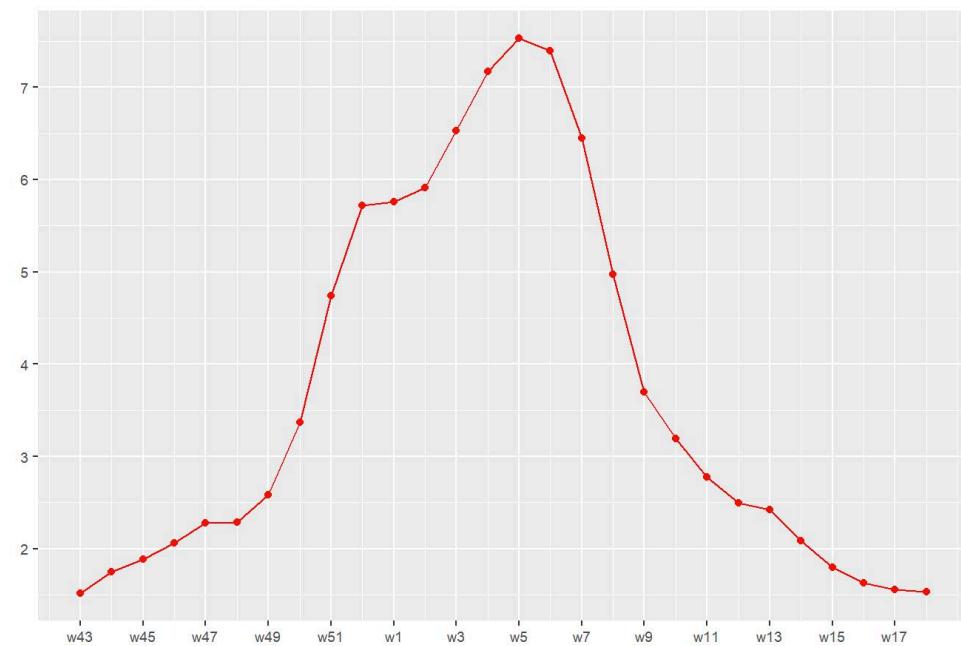
The Reich Lab) KCDE and Delphi's Bayesian Delta Density

The Reich Lab) KCDE and Los Alamos' Dynamic Bayesian Model

Appendix:

```
functionForDifferentModels("KCDE", "CUBMA", "Reich_Lab", "ReichLab_kcde", "CUBMA", thefilepath, otkdetest")
```

True % Weighted ILI Values



Difference in Point Values



Wish this HTML file would load? [Well, it does!](#)

If you ever want to know how to mutate variables in dplyr, [this is the place](#). Working in a population of favorably Southern, Non-Tobacco using, Bachelor's Degree Holding, 30-year-old mothers, we're able to predict a positive linear correlation between average age of mother and average birth weight, as well as markedly low Southern birth rates contrasted with markedly high Western birth rates, our multi-linear model is a little less than linear because the qq-plot has heavy tails, giving us some predictive ability on average birth weight.

MLH - Welcome to Our Pod Website

hip @ Pod 2.1.2

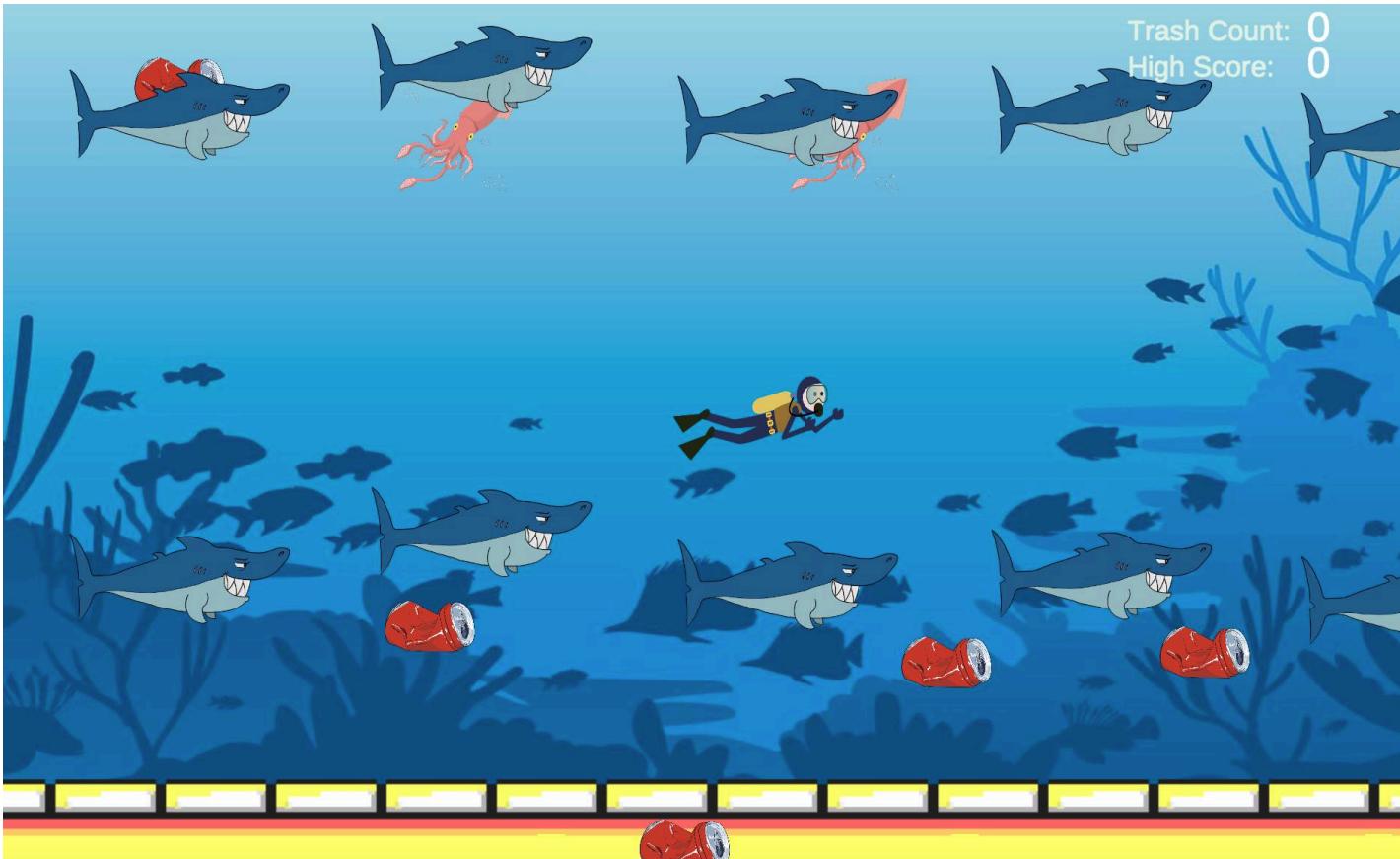


The most inspiring and surprising aspect of my MLH fellowship experience was building a portfolio website in React Bootstrap with SASS and plenty of other libraries, not to mention starting it myself. There are so many talented young ones out there who don't know that about themselves, that they could be the next generation of innovators. But what does it mean to be a really big innovator when [starting from scratch?](#)

Animating the snow effect and adding the hamburger menu among stylistic changes which were being concurrently added by the team (they were so good we don't need to call their name out) makes up for everything. I'm pleased to say that throughout, we maintained the attitude of maintaining a focus on everyone as an innovator, each with their own unique capabilities, with fondness for [playing those roles.](#)

Olaf's Swim

With the MLH Team



The cool thing about this game is that building scripts like GameAssets by referencing the Pipe object and setting script execution order, dynamically generating parameters across different C# scripts, all of this stuff is done in Unity. So although it looks like [the game assets](#) are in random locations which is a bit scary, they're actually linked together if you take a closer look.

And this game has seen enough C# to confidently show my comfort level on, say, a data science project involving C#. But enough analogies, [see for yourself!](#) I'll give you a couple pointers, you should have Unity installed in order to run the project from within the "Flappy Bird" folder. Creating superclasses and classes which instantiate and share game data, animating the scene with C# and game assets and sprites, following [this tutorial](#) and my teammates finding the pixel art for making prefabs, this project is really about learning [the Unity engine and rigid body](#), debugging with mouse logs of the current game instance, and putting C# classes together.

Visual Studio Code GitHub Projects.

Manage GitHub Projects from within VS Code!

The screenshot shows the Visual Studio Code interface with the GitHub Projects extension. On the left, there's a sidebar titled "Filter Projects" with checkboxes for "Personal Profile", "Repository", "Organization", and "Include Closed Projects", all of which are checked. The main area displays a list of GitHub projects under the repository "stackathon". The projects listed are:

- Stackathon Project Outline
- Fullstack Academy
- JTLL
- FullStack Foundations
- Fullstack Cyber Curriculum (Master)
- Fullstack Web Properties
- MLH Fellowship
- Relay Interactive Docs and Playground
- 0.5.X Python Projects
- Keras_Timeseries_LSTM_weather
- React Concurrent Mode Profiler
- React Native API Docs
- OhMyZsh-Theme Wiki Update
- Operation Cerberus

At the bottom of the sidebar, there are tabs for "PROBLEMS", "OUTPUT", "TERMINAL", and "DEBUG CONSOLE".

What about other ways of working with GitHub Projects? Luckily [we've got it](#).

The way that we did this was by compiling Svelte into TypeScript, working with Svelte which compiles into TypeScript files which decompile into JavaScript. The subtlety of it all is that this is an extension which fits neatly into VS Code without too much stigma because of its non-obtrusive nature; in fact the only reason this extension exists is that ever since VS Code GitHub Projects has become so versatile, working with Git beyond VS Code just seems like too much for a lot of novice programmers. But there's no need to write about it, see for yourself. We enjoyed making this as an alternative to the traditional Kanban board for collaborating on projects separately from GitHub Projects.

[Visual Studio Marketplace Link](#) 🍎.

Honestly, [I love my group](#) in their open-source dedication.

Fun Facts Website: Request Your Own Fun Facts.

Satisfy your curiosity about Cats!

Made with [the Cat Facts API!](#)

acts page,

anything you want to know.

A text message every day if you want,



-Dean Gladish

This project uses the [Cat Facts API!](#)

This is one of the most interesting things you'll ever see, what we've got is a Sweetalert which takes the number of requested facts (user input) and turns it into an Axios GET request to <https://catfact.ninja/fact> which calls the generateFunFact function [powering the site!](#)

Built on a foundation of Axios and React Bootstrap, my site is one of the most whimsical and fun places to be if you're looking for the best of create-react-app and a straightforward API request URL, the front-end is splendid.

Qooley App For Company Parties

The customer-side app for your kickoff annual meetings! When you want an artist, we're here! We've got 20 pages of activity booking. Let's get started on the introductory flow.

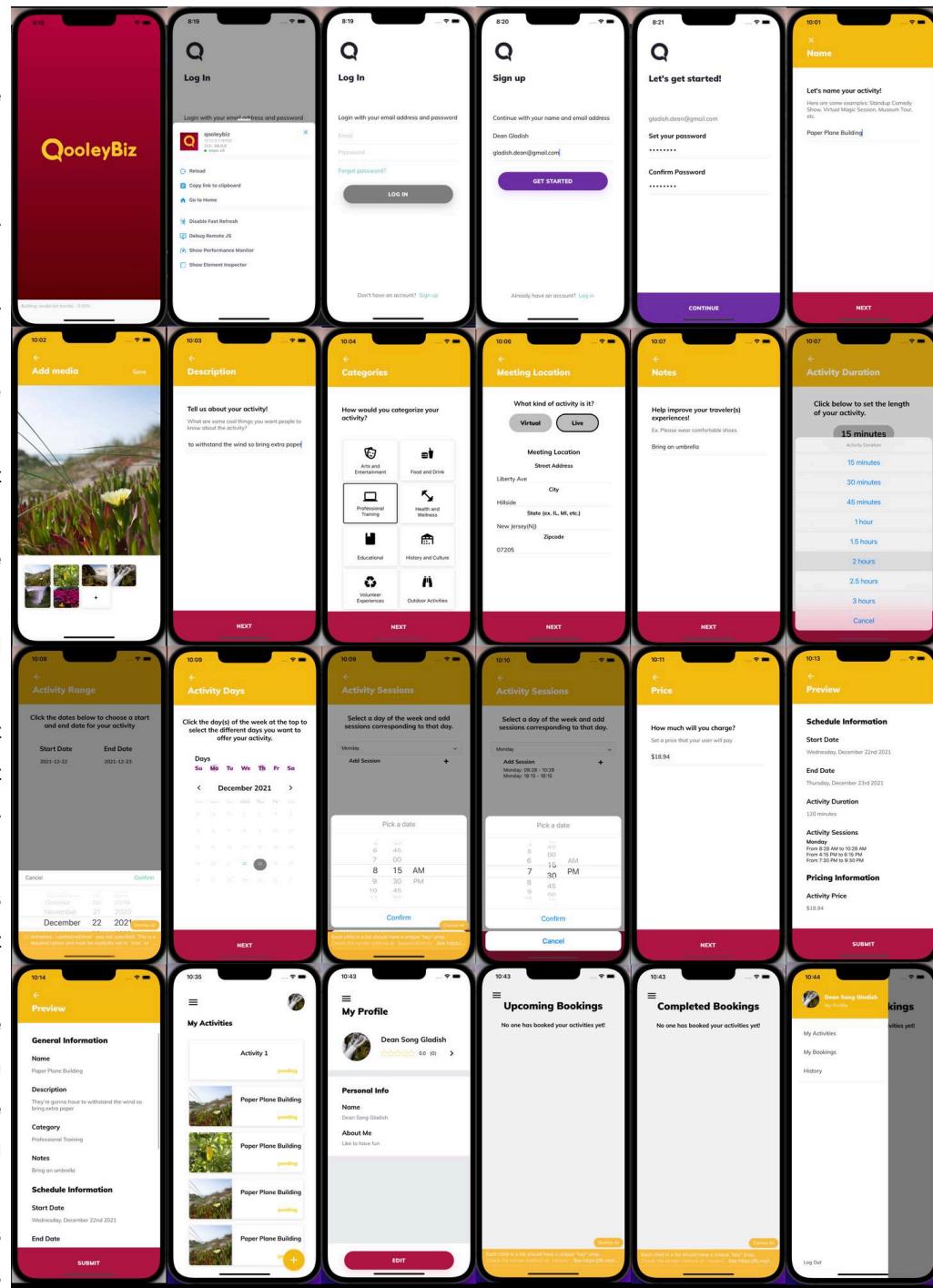
With React Native (react-native-bcrypt, react-native-calendars, react-native-credit-card), Expo (expo-font, expo-constants, expo-firebase-recaptcha, expo-google-app-auth), Firebase, and Lodash Clonedep, this app is perfect for company parties because we've got Stripe integration and an Express.js-powered backend for which, as seen on Qooleybiz, React Navigation allows us to create login forms with Axios, sharing the same relational database with the website except the app payment flow through Stripe.

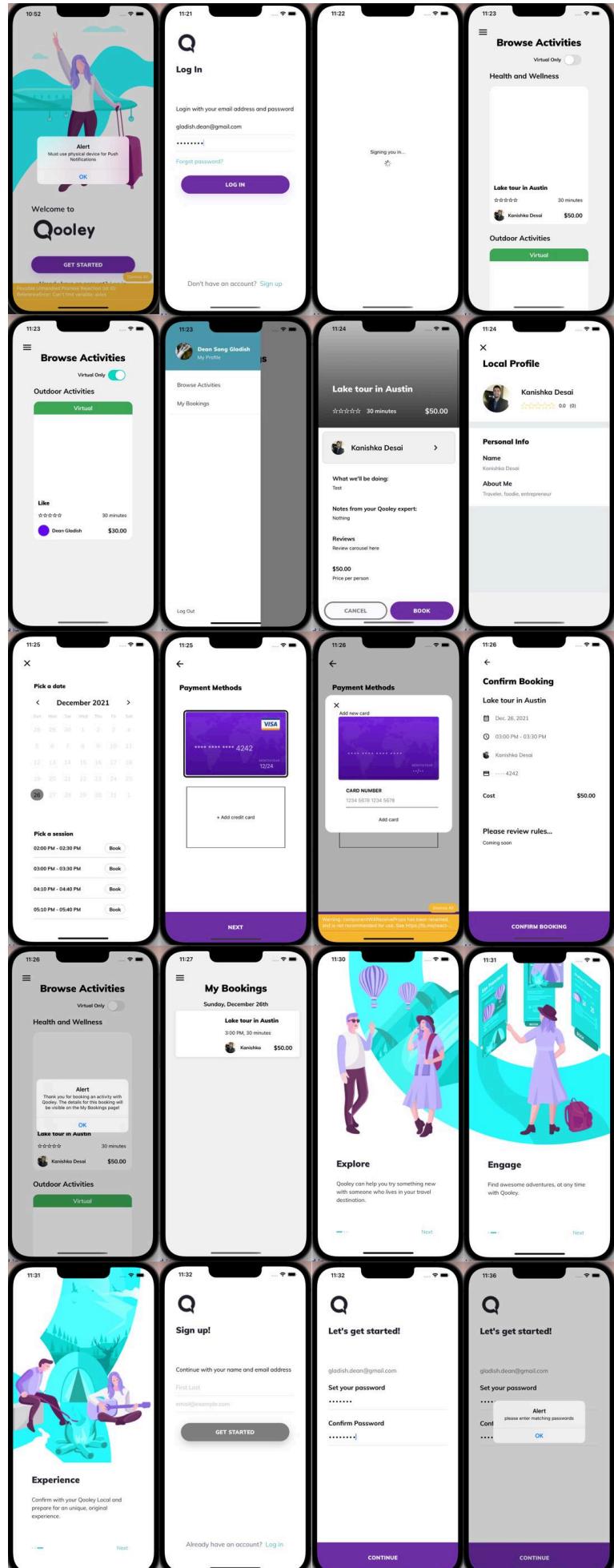
Feel free to reach out for a tour, expo publish and build and firebase deploy for DevOps is really exciting.

The Qooley App for Customers focuses on React Native props, placeholders for as-of-yet nonexistent user data on the prop tree, and the use of SDK 38.0.0->v39.0.0 to comply with the requirements of Expo Go. QooleyBiz has been fascinating how small changes live on the simulator can be pushed to TestFlight which is the App Store. We got the EAS Priority Plan, Xcode, we've got it all. While the Qooley App is the customer side (when you open the app it'll be on Destination Where are you going, Hanna?), the QooleyBiz app faces the experts, the local artists and experts. Here, we can sort by price distance or rating, date selector highlights like AirBnB, with future plans for Google Maps super simple integration, as well as integration with the host's Zoom or Google Meet!

When adding to calendar, don't cancel within 48 hours until the Activity, at which point you can begin to directly message or call the local expert; My Bookings shows upcoming bookings, and while Interests & Languages, Contact Info & Password haven't been implemented and the Name & About Me shows up on the Qooley Local side, Reviews and History go both ways; in fact, you can write reviews for activities and reviews for the experts; there are three different review tables.

The best thing is the flow - new customers can sign in, skip Google and Facebook, view their bookings in more detail, browse Activities, book bookings, payment and cancellation workbook, and should be able to cancel activity - that's the MVP. Watching Expo Turtle Queues for our build among many, expo publish the bundles index.ios.js and index.android.js with expo build:ios asking you to log in with Apple ID, I'm more interested in accessing EAS Priority Features in order to eas submit --platform iOS which opens command prompts for archive (deploy the build to the store) or simulator (run the build on a simulator). Since differences in build numbers represent recompilation of the same source (suppose it takes 542 builds to get to a 1.0.0 release), and expo version installation is so important for preventing EADDRINUSE. Topics of exploration are response body format, Crypt, search





algorithms for database queries, and our use of the Stripe API, on our Express API routes via req.session's stripe customer id.

QooleyBiz App for Artists

Tech companies, culture, and local tidbits!
The app for artists to host.

With the relational database collection app_users which connects to activities, what's not to relish? Our [developer program membership expired](#) so the MVP of the QooleyBiz app isn't online in the meantime. The QooleyBiz app is for the end users who want to make some money, while the Qooley app on the next page is for the end users.

We're searching and designing pop-up activity approval messages, and approval via admin portal, and we're most proud of Sohan's non-overlapping time slots. It's the best app for learning so much about JavaScript, like for example the fact that JavaScript for-loops recalculate the upper bound along with i (variable change), how to write firebaseConfig, Express API routes and git diff, unused isCustomer and isLocal boolean variables, catching edge cases and missing data versus making it mandatory (we made profilePic mandatory), Google SignUp, generating Firebase credential object with accessToken and data.profileObj, Sweetalert swal, error throwing on both passwords confirmation, Firebase authentication and user with email & password creation axios.post. We love Swal because it can be done in an else block, not just try/catch error blocks; for this we detect syntactical validity of HTTP request message framing and status codes. The Firebase API has .create User With Email And Password which should also have should have .delete User With Email And Password. My favorite thing has been field validation, referencing req.body.data to db.collection ('users').doc(user.email).

This is the artist-side rendering. Check out our next app on the following pages!

Welcome to My Blog

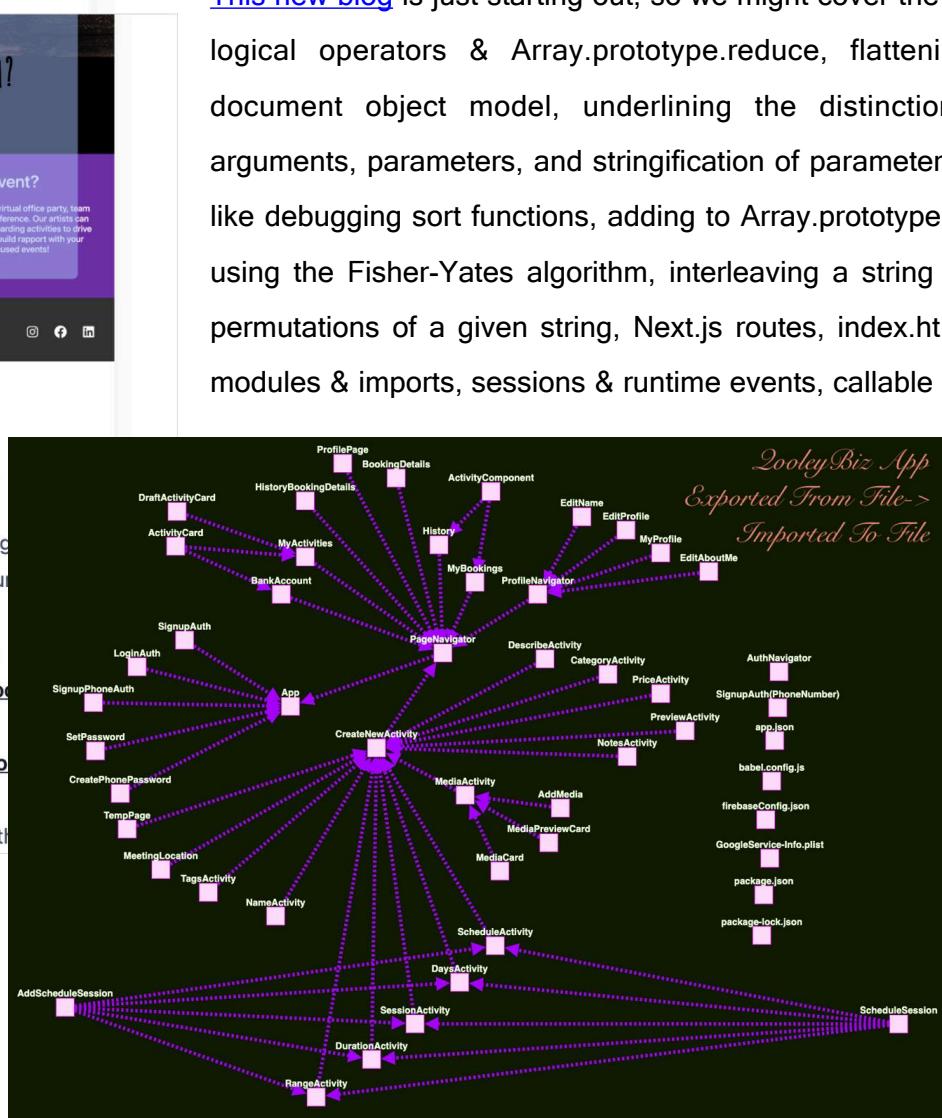
sv(sys.argv[3])

In the beginning during my first time building a blog in Sanity.io, using [the Sanity version of create-react-app](#) with [the sanity.io walkthrough](#) has been pretty fun. Setting up the Sanity CLI, adding CORS origins to the project API, and hosting with the Sanity CI/CD flow in Express.js and React Router has been one of the most rewarding experiences. [Front-end deployment using Vercel.app](#) and designing posts [in the studio](#) has been educational, given the extent to which React integrates with Sanity's base, components, core, default-layout, default-login, desk-tool, and vision modules. That's just the tech stack!

The blog started out on [Medium.com](#) writing about topics including data visualization in React using Chart.js, Celéri Health, React Lifecycle Methods and bi-directional communication with Sockets, Qooley.com, arrow functions in JavaScript, for-in versus for-of loop on properties of an array & iterating through non-array objects, insertion sort and bubble sort, dot and bracket notation in ES6, Array prototype methods, template literals, combining line breaks in long return statements & string interpolation. Don't hesitate to join in.

[This new blog](#) is just starting out, so we might cover the syntax and ES6 data structures like logical operators & Array.prototype.reduce, flattening multi-dimensional arrays, the document object model, underlining the distinction between variables, properties, arguments, parameters, and stringification of parameterized objects. We could cover topics like debugging sort functions, adding to Array.prototype, trapping rain water, array shuffling using the Fisher-Yates algorithm, interleaving a string of two other strings, generating all permutations of a given string, Next.js routes, index.html and jQuery with Webshim, CSS modules & imports, sessions & runtime events, callable and constructible functions, browser

scrolling, the class keyword, and micro-format class names. Benefits of concurrency for Node.js, require/import, shell scripting, event bubbling, and the AWS service provider for eCommerce. There's delight to be had soon in LeetCode problems like Trapping Rain Water, Burst Balloons, Decode Ways, Reaching Points, Minimum Window Substring, and Text Justification.



Internet Radio Streaming

With the Fullstack Academy of Code title and the MIT License on GitHub, and the Radio function itself in the radio.js component which adds an event listener to the user-selected radio station text div using the howl attribute of the newly generated Radio function-object with radio sources set to IP addresses of BBC



89.5 Radio 1
90.2 Radio 2
91.8 Radio 3
94.8 Radio 4
103.3 Radio 90

and Absolute radio streaming URLs, we're able to exhibit Howler.js functions in a neat web box.

This site is created and included in this portfolio because, having learned about Howler.js at Fullstack Academy in order to set it out, learning how easy it is to play back audio has been a breeze. Thank you for listening and feel free to browse [on the site](#) because it's a brilliant topic, internet radio streaming is.

Movie Reviews & Topics



CITY HALL

Topic: City Hall is one of the few films Wiseman has shot in his hometown, Boston, not that you'd necessarily know about that connection from the film.
Producer: Frederick Wiseman
Premiered: 8 September 2020
Average Rating: 78.9%



CITY HALL

+
Topic: City Hall is one of the few films Wiseman has shot in his hometown, Boston, not that you'd necessarily know about that connection from the film.
Producer: Frederick Wiseman
Premiered: 8 September 2020
Average Rating: 78.9%



TYPESCRIPT

Topic: TypeScript is an open-source language which builds on JavaScript world's most used tools, by adding static type definitions.
Producer: Microsoft
Premiered: 1 October 2012
Average Rating: 75.4%



TYPESCRIPT

-
Topic: TypeScript is an open-source language which builds on JavaScript, one of the world's most used tools, by adding static type definitions.
Producer: Microsoft
Premiered: 1 October 2012
Average Rating: 75.4%
Rating 0: 98%
Rating 1: 76%
Rating 2: 98%
Rating 3: 76%
Rating 4: 89%
Rating 5: 68%
Rating 6: 10%
Rating 7: 67%
Rating 8: 97%

open-source

despite its name, TypeScript doesn't have proper type validation

add tag for portfolio

Add a tag



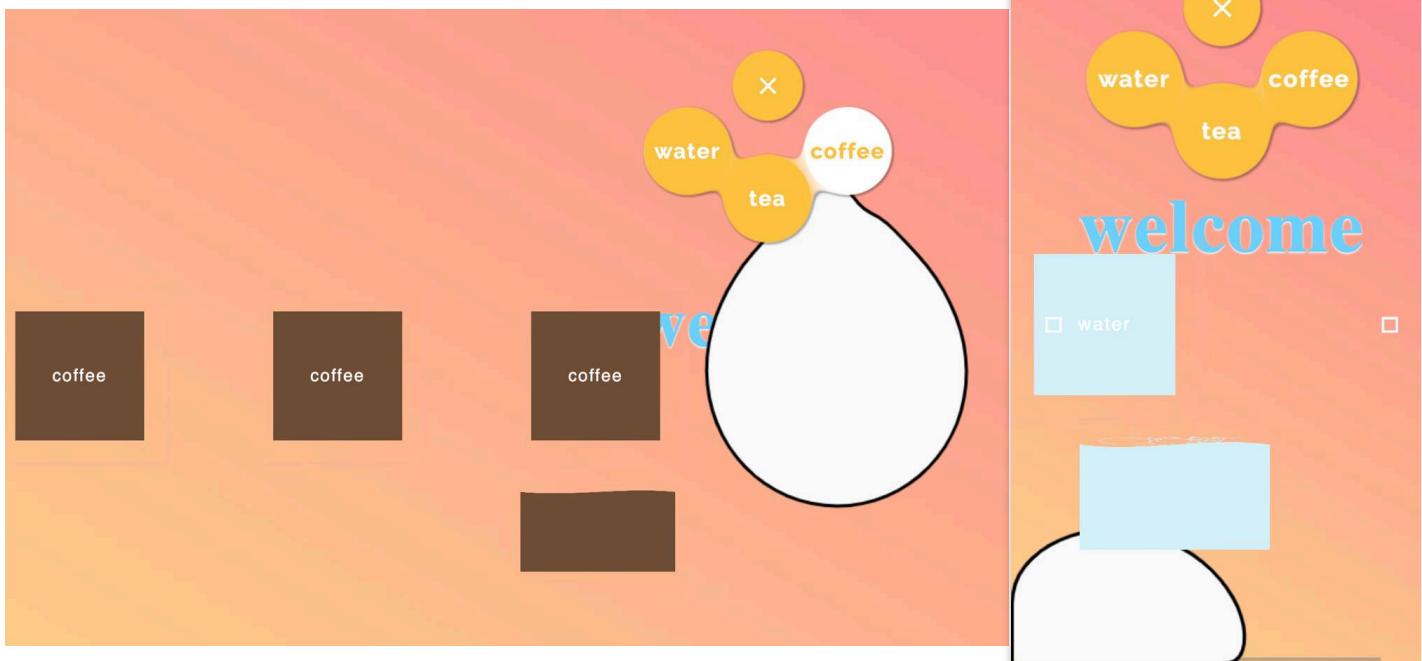
THE WITCHES

Topic: It's an adaptation of the 1983 Roald Dahl book about a kid and his grandmother who get caught up in the dastardly plans of a group of chi witches at a seaside hotel.
Producer: Nicolas Roeg
Premiered: 22 October 2020
Average Rating: 81.8%

One of the most exciting beautiful UI examples that you've ever seen. The one that's all front-end, so while one ordinarily works with PostgreSQL for data-oriented sites like Chalkboard, JDB Cars, and [gooley.com](#) with the apps, all the tags are stored locally on the browser while the App.js component contains our data object.

It's a two-file application (one file to concatenate the items, and one file to describe a single movie entry). That means that the CSS contains all the borders, box-shadows, margins, heights, background-colors, are all calibrated for the media mobile screen with maximum width 600px. With regard to CSS this site is about setting ::placeholder, :-ms-input-placeholder, and ::-ms-input-placeholder for different web browsers like Chrome, Firefox, Opera, Safari 10.1, Internet Explorer 10-11, and Microsoft Edge, and that's actually one of the biggest things to learn how to do; testing on different versions, working with software engineers and being a phenomenal web developer means being universal with every web browser. [There's also a mobile view!](#)

Coffee in CSS3



The [most interesting product coffeemakers](#), dreamt up on a foundation of

Node.js, Webpack, demonstrates working knowledge of color schemes in CSS3 like box-shadow on React Carousel cards, and different screen size portrayals of the liquid bubbling up. Includes rattling for each coffee maker and document properties of the canvas. See if you can visualize the react-multi-carousel!

The draggable blob, water dripping, gooey menu, display a knowledge of SCSS which animates each blob by number and keyframe, with react-script-tag to append various styles to DOM elements, trying all kinds of stuff like vue-style-loader, postcss, mathsass, compass, and math-helpers for the gooey menu. This demonstrates an excellent @media mobile optimization with webkit, moz, o, and ms transformations in the form of rotations, @including dynamic @mixin nth-child positions for animating the blob, and keyframes describing the movement of liquid drops from higher heights to lower heights.

This is a portrayal of the greatest animation, wave 1s ease-out forwards with feGaussianBlur and feColorMatrix for the water drops, componentized draggable blob, gooey menu, liquid bubbles as the invisible glass fills, React multi carousel, and cubic-bezier animation for each water drop. My favorite part is putting all animation settings in SCSS on one line, working with z-index to layer the background with the blob and coffeemaker menu selector.

Chess Site

This site is the milestone of my interest in board games like chess; starting out on the front-end with the tutorial for [developing a two-player chess game with React.js](#) which is bootstrapped with create-react-app and introduces piece prototype functions, and then successfully doing the two-player chess game has been the most rewarding experience because during testing, there's a chance to take off with debugging features like deleting square colors by

commenting out the non-configurability of the square objects, making sure the background color is persistent when you move pieces and on the other hand, fixing the error where we click an invalid destination for a selected piece and the selected piece stays highlighted. Socket I/O, which is what we learned at Fullstack Academy as our primary method of building a WebRTC-esque application and then on top of that importing \$ from jQuery which automatically scrolls the chatbox to the bottom whenever people add or receive messages in the hatbox, clearing the text field on submit and successfully getting the number of people online to display, passing system time down through ReactDOM.render to be rendered on the React component, are all phenomenons which you shall see in [just a moment](#).

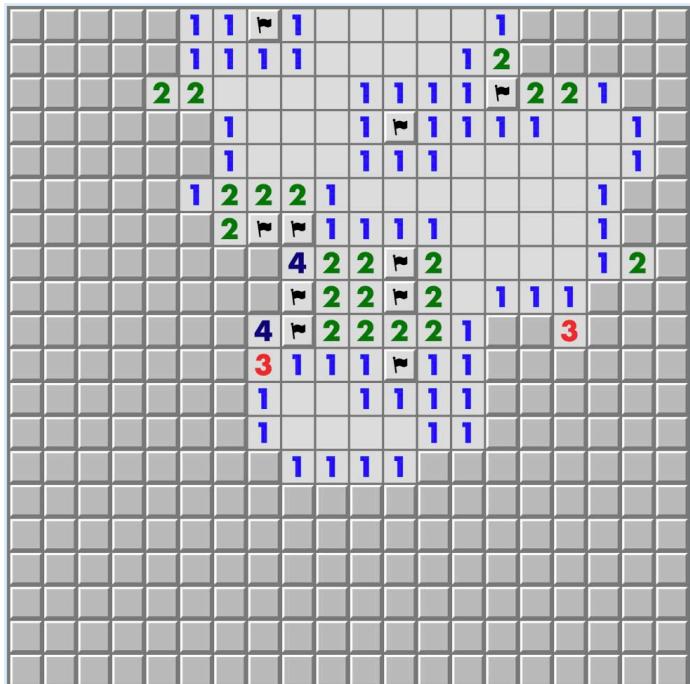
The screenshot shows a chessboard with 64 squares. The board is set up with white pieces on light squares and black pieces on dark squares. A piece has just moved from e4 to e5, as indicated by the text "Moved to e5." in orange at the bottom left. The sidebar on the right displays player information: "dean Player 2 (29 minutes 54 seconds)" and "White Player 1 (29 minutes 46 seconds)". It also shows "Online Players: 1" and a "Moves" section with the first move "1. e6 e5". The session log at the bottom shows activity: "New game room created. just now", "Player 1 connected. just now", and "Player 2 connected. just now". There is also a text input field with "Enter" placeholder text.

Discoveries include the fact that Heroku and jQuery mesh only without the React use effect, converting PGN notation to an array which is then stringified onto our database, getting different URL slugs for respective game rooms, passing game room ID stored on the URL slug down to the Game component, fixing non-unique key error when rendering squares, and adding user authentication just using sessions so that people can be spectators or play games wherein the frontend-backend accesses the session ID through Redux successfully, while selectively retrieving the game data stringified in the database. Regardless of the Spectate/Player 1/Player 2 options which are set up depending on the number of players

registered for a particular game room based on their session id, looking into handling unload (so that when the user leaves the webpage the server performs certain tasks such as modifying (PUT/DELETE) game data) has been one of the most rewarding experiences that I have ever had. Actually, Express connect.sid (session id) is now being stored on the database when you select a game room and being removed from the database when you unload the page; the piece prototype functions must be re-added, given that in this scenario when you do JSON.stringify (and then add the string to the database) the piece prototype functions are removed.

With the “it’s not your turn!” thing going, players can move because I’m re-creating the board every time based on what’s in the database (the game object is being stored in the games table of the db). Originally, three io connections (Socket I/O connections) are being made every time a single person connects to the website. Every time someone makes a move, the data is sent to the database, always starting on white (player 1)’s move when it really should alternate, which in my view was an issue and so now the turn is also stored in the database; we want to make sure that every time a move is made (the component updates), this is reflected on both players’ screens. Completing bi-direction communication and saving the

time-gap delay (one move isn’t enough; two moves shows player 2 the first of player 1’s moves) by Redux allows whose turn and which player is next to be stored in the database, reduced on the Redux state and therefore registering properly. Players are restricted to play only their turn, tracing the socket flow allows chatters to successfully see new messages from other clients without having to refresh the page.



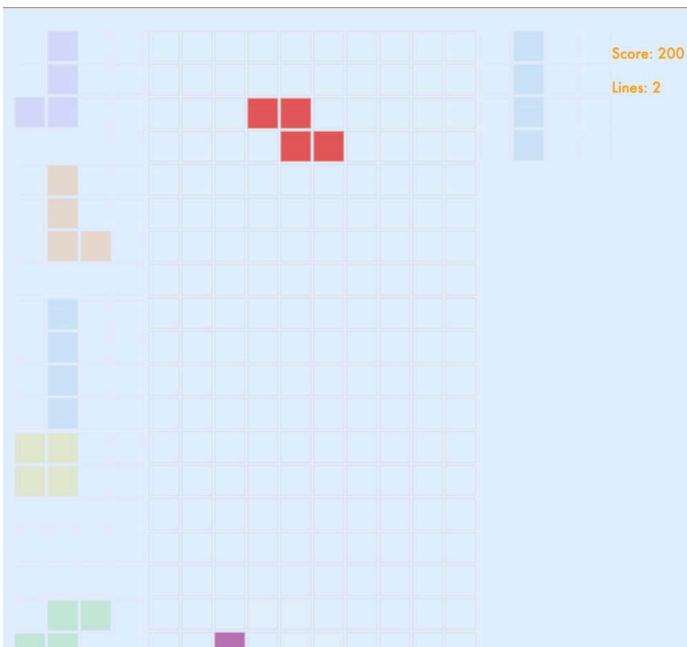
With regard to sockets, sockets tell the client that it's time to do something to the database. Then, the client determines if I'm player X and I just made a

move, stop the countdown and PUT my new time to the database. If I'm player X and it's now my turn, restart the countdown.

React provides the framework in place to pass down props through socket.emit whenever we need to,

say, modify the time stored in the database or record the specific move that is passed to the putGameThunk at that particular moment. Furthermore we have the framework to start and stop timers - the time remaining should be stored in the database table.

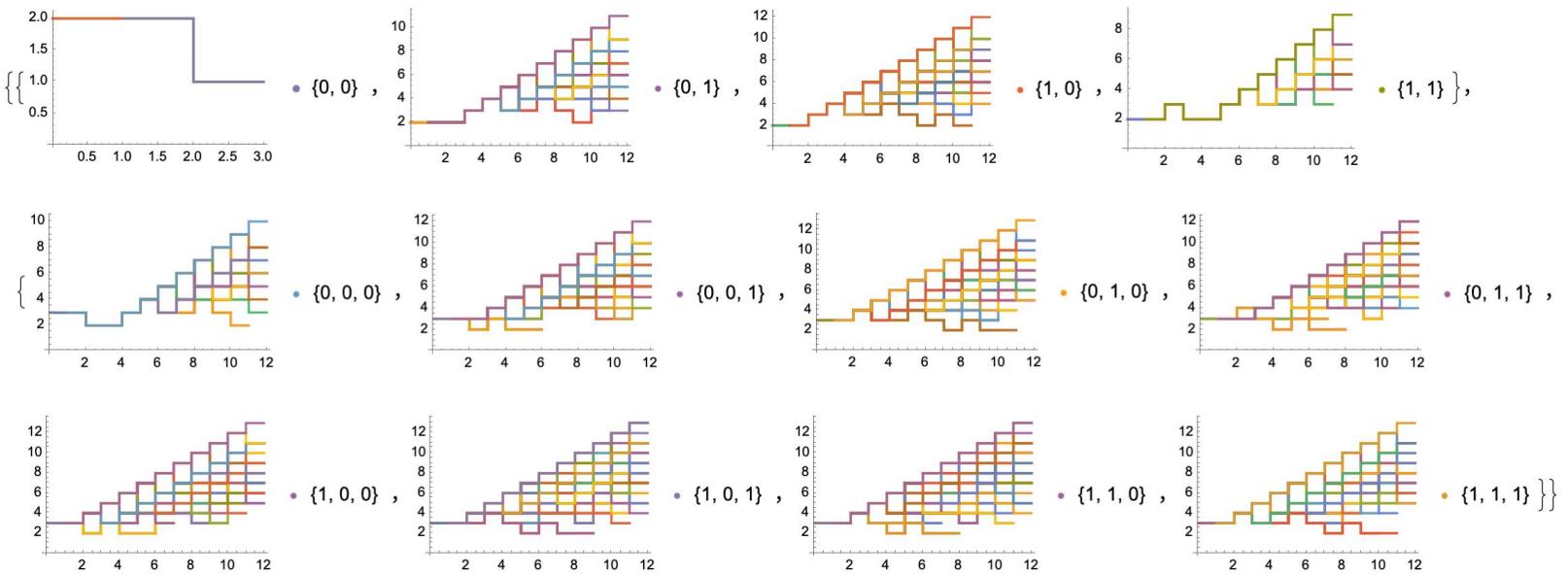
If you like Redux then you're going to enjoy what this site implements in Redux, you basically need to use it if you want to refresh your components. If you don't want to refresh your components because doing so creates an infinite loop when you're trying to deal with PUT requests to the database, to update the time remaining for player 1 or 2, then you should actually not use it in that specific use case and only edit the database, not the Redux store. So, the updated time from the reducer is not getting retrieved from the database. Actually every time someone makes a move the data refetches and the timers make PUT requests with a criterion which prevents multiple requests from being made, integrated automatically into the API route. Timer, moves, and online players are all fixed. New player registrations are displayed on the database and client-side!, not reflected on player 1's side until they make a move.



Changing the rules and adding Tetris, POSTing nicknames and time limits to the database, detecting mouse downs and mouse ups for which mouse down highlights possible moves down the event chain, and highlighting squares when you hold the mouse down... I'm honored to mention.

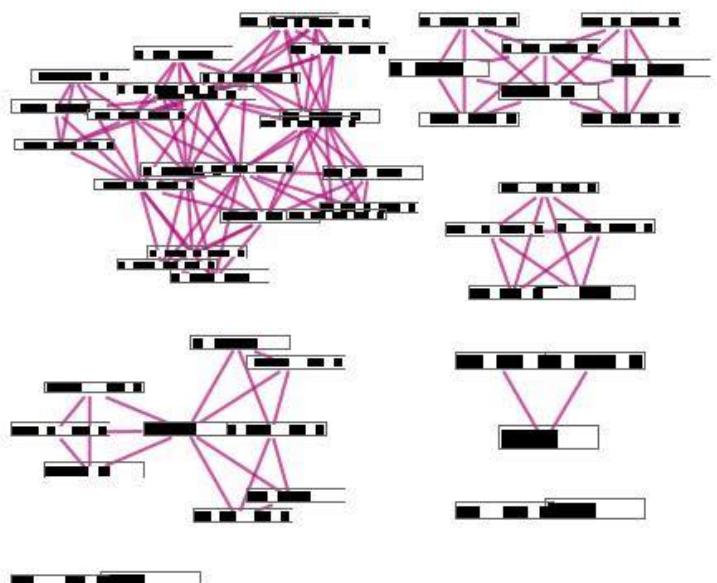
Debugging has been a wild ride for which the session id depends on setting a nickname and time limit at the beginning, fixing duplicate piece glitches and removing generate sourcemap = false from package.json, adding build scripts, and removing interference in deployment via the on-build script removal, and adding the nabber which you see at the top is an excellent display of the game logic. The two images which you see of Tetris and Minesweeper I'm integrating separately through the react-tetris and react-minesweeper libraries respectively.

Wolfram Summer School 2022-Adventures in Multi-way Computation.



The Adventures in Multi-way Computation that made it all possible: at Wolfram Summer School 2022, we developed an article on [Multi-way Tag Systems in Coordinated Rulial Space](#).

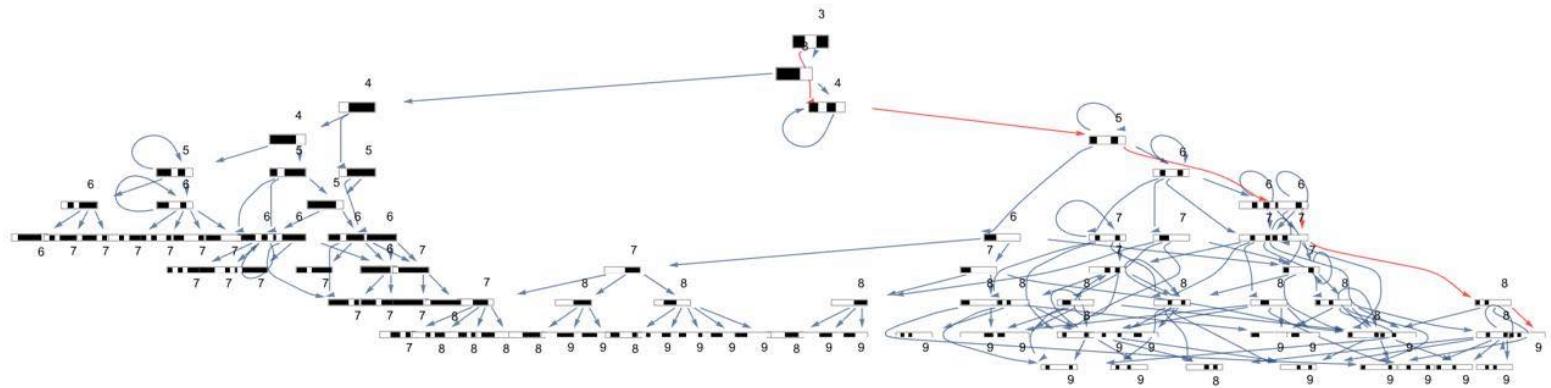
That's the official paper which explains how probability distributions stabilize, that is become static over time steps in the causal event graph, in Wolfram Language.



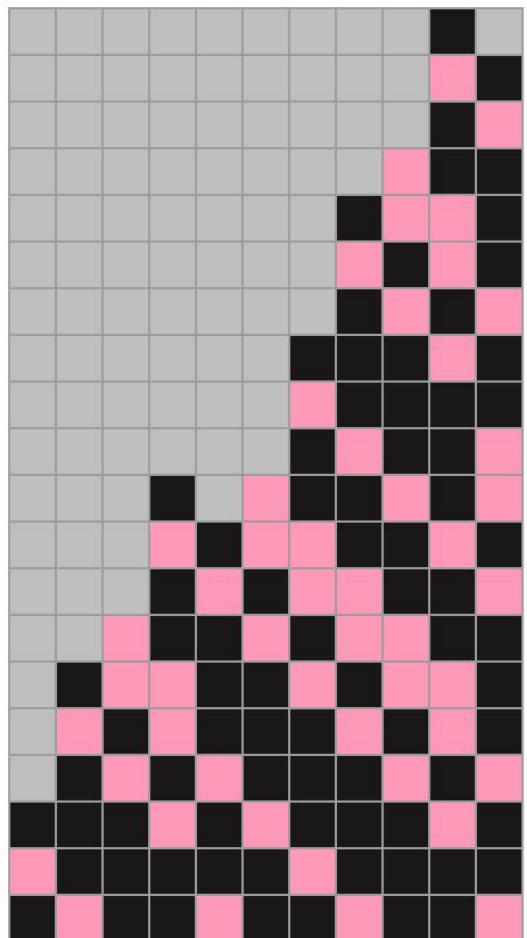
With friends and colleagues from around the world we visualize computational complexity via generational string re-writing in Rulial space across all multi-verse orders to understand probability decision distributions, symbolize prime computational irreducibility, and contribute to the Research & Development operation that Stephen has become reasonably good at (he even takes the time out of his day to, although he doesn't guarantee every line of code, comb through the function repositories

like MultiwaySystem that are submitted so that our code is vetted). It was like the stock trading floor in New York City or the patent office in Manhattan.

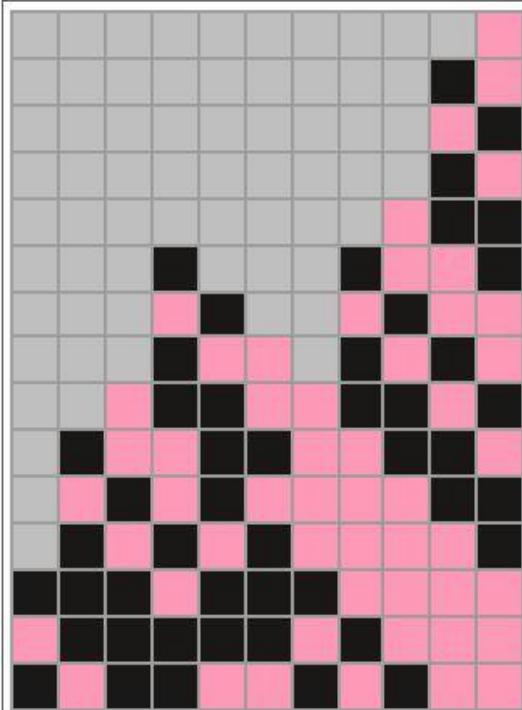
My interest in modern computing, the profound mathematical knowledge with no socially constructed basis, and the hardest Y Combinator to solve, made WSS22 a really fun exploration into the law of excluded middle, universal computation, hyper graph re-writing, the isomorphisms of meta-mathematical space, the non-computational, the quantum interactions that make traditional math and physics as we know them, possible.



```
{left___, 0, 0, s___} → {0, 0, 0}
{left___, 1, 0, s___} → {0, 1, 0}
{left___, 0, 1, s___} → {1, 0}
left 1 1 s 1 → {1, 1, 1}
```



Learning about Mathematica was so refreshing; lectures from industry leaders like Paul and Bob were eye-opening and everyone was so engaged that by the time we did book signing and poster & article presentations we were super pumped to get home and read them all.



Guessing Game.

One of my favorite projects from Fullstack Academy, The Guessing Game is a nice introduction to CSS and HTML. Aside from asking questions about how to improve [my website](#) it allowed me to learn about



GitHub Pages, CSS3, HTML5 selectors, IDs, and classes. It was a proper introduction to how IDs can be unique and have special functionality, and how to define how elements behave via CSS properties. The HTML aspect is also quite complicated.

In the [JavaScript](#), the first half of the file (I did this as a single page file) is about the display and how it looks in that it uses the Fisher-Yates shuffle and the class keyword that has been described as syntactic sugar. The second half of the file is about detecting user input, through events. My understanding of jQuery and the features it adds is that jQuery uses a different syntax.

If you want to focus on a specific element then the Document interface allows us to change that element's text, via the Document properties and query selectors that return a string. The background attribute of the page provides the enter functionality and specific button references, for which the already-guessed numbers need to be displayed.

The hints are accessible with winning and losing included, and there is user input focusing for all buttons. The hints are close together, and the font has been changed. The theme is set to jekyll-theme-cayman, Fullstack Academy is in the footer, and dark mode is toggle-able. The logic and the CSS is neatly organized.

The Mangrove.

The prototypical research project for politics. I built this site with a childhood friend named Andrew, and if you've ever picked up a ballot and seen dozens of names of people and positions you had no idea existed then you're not alone. Our site tries to provide easy-to-access information about local politicians, and our aim is to provide more information on what or who we are voting for. The idea is that instead of say, making multiple searches to find out who they are you could find out more about your local elected officials by browsing [our site](#). Try it, the username for testing is loginthemangrove@gmail.com with password themangrove.

[SHOW USERS](#) [LOGOUT](#)

1029 Main Street, Pittsburgh, PA 15215

QUERY GEOCODIO LAT/LNG

38.9002898, -76.9990361

QUERY GEOCODIO ADDRESS



Name:
Carrie DelRosso
Party:
Republican
Position:
State Representative



Name:
Lindsey Williams
Party:
Democratic
Position:
State Senator



Name: Chris Deluzio
Party: Democrat
Position: Representative in Congress

The raison d'être is to revolutionize the political system, grant voters easy access to information about their local and state-level politicians, and connect politicians to their local communities via progress markers, trackers, and the information that provides the accountability we absolutely want after the great Centennial (40% of startups in Y-Combinator are purely in the idea phase!).

Designing this site has been phenomenal, figuring out how to implement the logic in login/signup templates to [what we have](#), merging GeoCodio API queries with MongoDB Access Control collections. Our current UI is great for now, and the login & signup pages have been wonky and fun to [write about](#). Altogether it is a fun project, and I look forward to joining you on your programming path.

■ Contact me at dean.gladish@gmail.com or gladish.dean@gmail.com.

■ Check out [my tutorial](#) on Jekyll Template Website Creation!

■ My phone number is (224) 306-4486.

■ Or you can follow me on GitHub at <https://github.com/gladishd>.

■ LinkedIn is at <https://www.linkedin.com/in/dean-gladish/>.

■ Credit goes to the <https://www.template.net/editable/62157/computer-technology-magazine> creators and then perhaps most of all Kanishka for introducing all the artists, speakers, performers, entertainers, comedians, musicians, coaches, and finally company packages for our website Qooley.com, the educational project envisioned by the one and only Kanishka Desai!

■ Check out my [Wizard News Development with Express](#), [PostgreSQL TripPlanner](#), [WikiStack forum](#), [Color Picker](#), [JukeBox](#), and [Adoption Centers](#) video (and React DevTools on [my Adoption Centers project](#))!