

Electronic Golf League Scheduler

Ethan Evans, Aidan Andreas, Brady Zalasky, Nick Landon, Maxwell Farver

Team: sddec21-03

Website: <https://sddec21-03.sd.ece.iastate.edu>

Advisor: Mai Zheng

Client: Tina Prouty



Project Overview

Our Client

- Tina Prouty
 - Academic Advisor for E CPE
- She administers a women's golf league
 - Honey Creek Golf Club in Boone, Iowa
- Looking for a more modern and robust system
- The club size outgrew the current system
- Looking for an affordable and user friendly service



Overview of Project Solution

- The old system was an excel spreadsheet
 - Not convenient or modular
- The system being developed is a web application
 - Fast-rendering and responsive for multiple devices
- Features to be implemented:
 - Enter team members names
 - Create hole assignments, if absence golfer must be replaced
 - Leaderboard
- No other applications that are free or low cost

Project Requirements

- Functional
 - Hole assignments, schedule, team pairings, leaderboard
- Economic
 - Low cost, minimal monthly or no monthly cost
- User Interface
 - Simple, features are easy to use, design is modern and professional
 - No constraints on design choices such as color or style
- Version Control
- Software Testing
 - Implement user interface and unit testing
- Security

Team Member Responsibilities

- Each member would like experience with front-end and back-end
- Development will be broken down into features
 - Each member will take either a front-end or back-end role for the feature
 - Next feature members will switch their current role
- Allow for each member to broaden their skillset



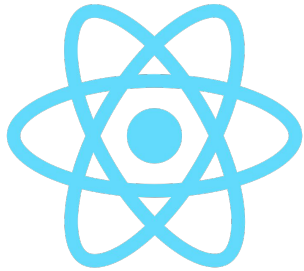
Application Design



Design Decisions

- Web Application - Netlify and ReactJS
 - No need for installation or updating once the application is deployed
 - Works with all devices
 - Free hosting
- Microservices/serverless architecture - AWS Lambda and NodeJS
 - Keeps running costs low
 - Extensible by design
 - HTTP request validation handled automatically by AWS
 - Simple testing
 - Auto-provisioning of resources from AWS

Tech Stack



Tech Stack

- AWS chosen over Azure and Google Cloud Platform
 - Generous free tier of services
- DynamoDB chosen over other SQL and NoSQL options
 - AWS free tier
 - Low latency reads and writes ideal for serverless architecture
- NodeJS chosen over other Lambda runtimes
 - Speed during cold starts which will be the majority of interactions
 - Jest will be used for unit testing
- Netlify hosting chosen over AWS Cloudfront
 - Simplicity of deployment and domain configuration

Design Plan for Web Application

- Single Page Application written in React
- Each page will be in its own directory
 - Trivializes adding new pages
 - Isolation allows for easy testing of each page
- Interface components will split by functionality
 - Smaller components allowed for increased readability and easier testing
- Hosted with Netlify
 - Automatically deploy when a commit is pushed to the main branch

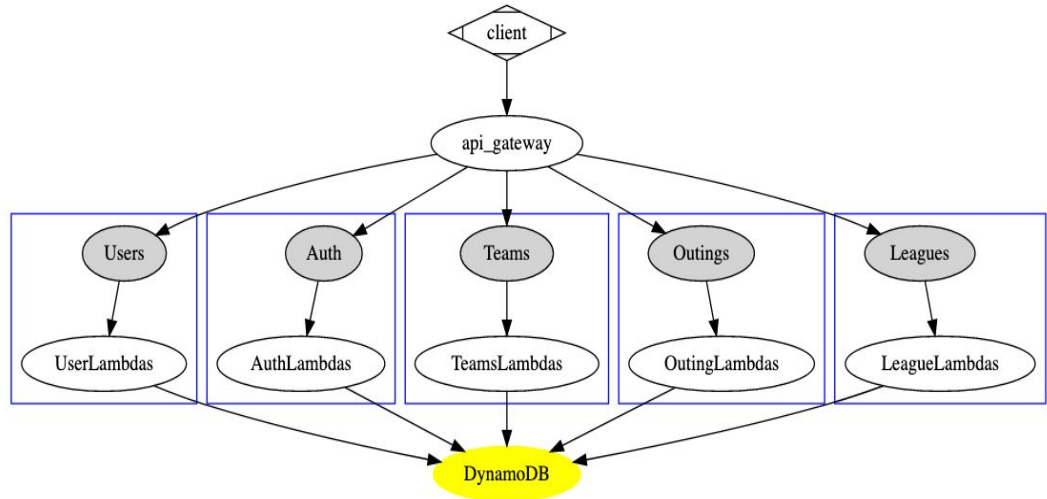
Design Plan for REST API

- Written in NodeJS
 - Each endpoint hosted on its own AWS Lambda instance
 - Each request will pass through an AWS API Gateway
- API Gateway will
 - Validate the request body and respond with an error as needed
 - Attach requestor's authorization credentials
 - Pass the request to correct Lambda endpoint
- Each service will have a DynamoDB database schema
 - DynamoDB is distributed by design so multiple instances won't be needed

Inside a Microservice

Path of a request:

1. Client sends HTTP request
2. API Gateway
3. Lambda
 - a. Communicate with other Lambda functions if needed
 - b. Access DynamoDB
4. Response sent back to Client

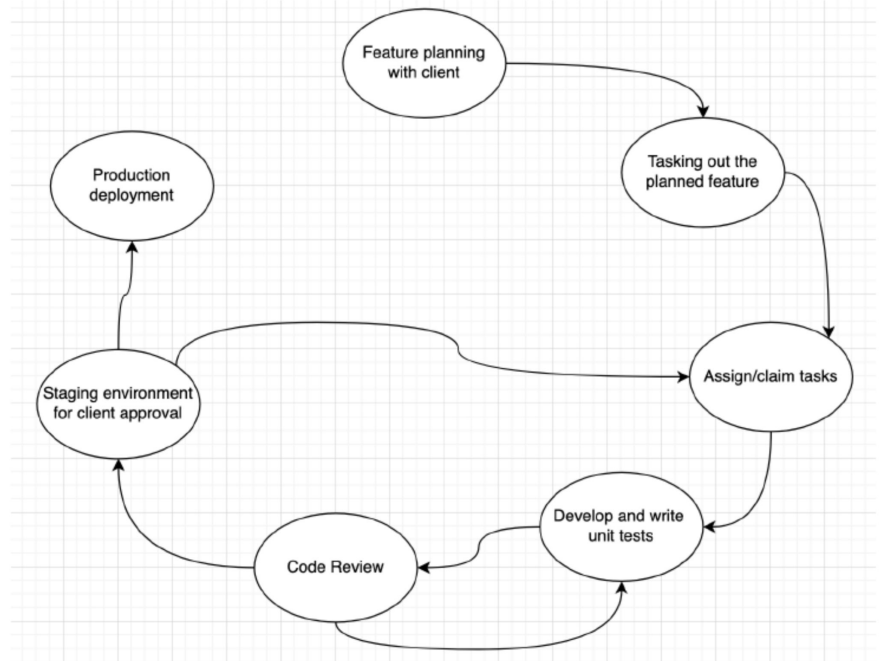


Testing

- We will be utilizing unit testing, interface testing, and acceptance testing
- Unit Testing
 - Executed within each microservice
 - Test all business logic and database interactions
- Interface Testing
 - Testing multiple services through HTTP requests
 - Completed after unit testing, mimicking real world use of the product
- Acceptance Testing
 - Features will be pushed to environment visible to client for verification

Development Process

- Agile Variation: Kanban
 - Continuous backlog
 - Non-traditional sprints
- Multiple quality checks for each feature:
 - Unit Tests
 - Code Review
 - Integration Tests
 - User Acceptance Testing



Progress

- Have met with client and settled on technical and design constraints
- Have planned out our design and buildout in depth
 - Our design doc will be useful as reference and road map for when we begin the build out
- Have created wireframes to give us a clear picture of what we want the finished product to look and feel like
- Have communicated with each other about how we plan to delegate tasks and responsibilities within the team
- Are ready to begin familiarizing ourselves with development tools and environments and begin the buildout

Preliminary Design

Golf Scheduler Assignments Schedule Teams Leaderboard Settings

Hole Assignments

Date Shown: **August 14th**

Hole	Team 1	Team 2		
1	Blue Team	Red Team	<input type="button" value="Update"/>	<input type="button" value="Delete"/>
2	Green Team	Yellow Team	<input type="button" value="Update"/>	<input type="button" value="Delete"/>
3	Purple Team	Orange Team	<input type="button" value="Update"/>	<input type="button" value="Delete"/>
4	Brown Team	Teal Team	<input type="button" value="Update"/>	<input type="button" value="Delete"/>

Golf Scheduler Assignments Schedule **Teams** Leaderboard Settings

Team Pairings

Team Name	Members		
<u>Blue Team</u>	Suzy Q	<input type="button" value="Update"/>	<input type="button" value="Delete"/>
	J. J.		
<u>Red Team</u>	Tina Pajak	<input type="button" value="Update"/>	<input type="button" value="Delete"/>
	Jane Doe		

Preliminary Designs - Cont.

Golf Scheduler Assignments **Schedule** Teams Leaderboard Settings

Upcoming Outings

Date	Time	Participants	Update	Cancel
8/14/2021	10:00am	32	Update	Cancel
8/21/2021	10:00am	28	Update	Cancel
8/28/2021	10:00am	25	Update	Cancel
8/4/2021	10:00am	30	Update	Cancel

+ Add New Team

Golf Scheduler Assignments Schedule Teams **Leaderboard** Settings

League Leaderboard

Currently Shown: **Cumulative**

Team	Score
1 Blue Team	-3
2 Red Team	-1
3 Orange Team	2
4 Teal Team	3
5 Yellow Team	5
6 Purple Team	8



Conclusion

Team Challenges

- Familiarity with development tools and environments
 - Most of the tools we are using are new to us and require some time to gain familiarity and understanding
- Communication with advisers and each other
 - Can be difficult to work around schedules to meet as a team and with advisers
- Developing a quality product that also adheres to constraints provided by client
- Any unexpected problems that may arise

Future Plans

- This semester has provided us with a good structure and plan to begin building off of
- We plan to start building out the site as soon as possible
- We plan to maintain frequent contact with our client and adviser in order to keep us on track and on the same page with each other
- We are excited to get to work and hope to deliver a quality product next semester

Comments/Questions?

