# SNEHITA KOLLI

**Phone:** 925-290-7317 | **Email:** snehitakdevelops@gmail.com
**LinkedIn:** https://www.linkedin.com/in/snehita-kolli-0abb23b1/ | **GitHub:** https://github.com/snehitak20
**Portfolio:** https://snehitak20.github.io/job-hunt/

## PROFESSIONAL SUMMARY

Future full-stack web developer leveraging a medical sciences background to build a more intuitive user experience on the web. Earning a certification in full stack web development from UC Berkeley Extension. Newly developed skills in JavaScript, CSS, and reactive web design, with extensive knowledge in medical and clinical terminology. Known as a dependable and organized candidate that is ready to learn and contribute to organization success. Passionate about developing applications that will engage users with an impactful user experience.

## EDUCATION

**Certificate in Full Stack Web Development,** 06/2022
UC Berkeley Extension, Coding Bootcamp
**Master of Science, Basic Medical Sciences,** 12/2021
St. George's University, School of Medicine (SGU)
**Bachelor of Science, Biology,** 05/2017
St. Mary's College of California

## TECHNICAL SKILLS

- Languages: MySQL, JavaScript, HTML5, CSS3
- Deployment and Delivery: Heroku, Command Line, GitHub Pages, Git
- Tools: Sequelize, Express.js, ES6, Node.js, OOP, Insomnia, JEST, Moment.js, jQuery, Bootstrap, local Storage, Chrome Development Tools
- API Design: AJAX, Fetch, JSON, Client-server model, HTTP request methods, Rest API, Server-side API, Third-party API, Relational Databases
- Microsoft Office Suite, Google Suite, & Adobe Suite

## PROJECTS

**Subscribe-o-Matic (May 2022)**

- An application where users can subscribe to their favorite NBA basketball team. Users will receive an email confirmation when subscribed to a basketball team. Users can access subscribed teams in their profile page. Deployed via Heroku.
- Technologies used: JavaScript, Express.js (Sequelize, dotenv, Express-Session, Connect-Session-Sequelize, Express-handlebars, bcrypt), Nodemailer, MySQL, CSS
- Link: https://subscribe-o-matic.herokuapp.com | GitHub: https://github.com/snehitak20/subscribe-o-matic

**E-Commerce Back End (May 2022)**

- The back end for an e-commerce application. Models, routes, and the main server are written. An Express.js API will use Sequelize to interact with a MYSQL database. API routes tested within Insomnia.
- Technologies used: JavaScript, Express.js (Sequelize, dotenv), MySQL, Insomnia Core
- GitHub: https://github.com/snehitak20/shopping-problems

**Employee Management System (May 2022)**

- A command line application to manage the company's database. Users can access a variety of options to view departments, roles, and employees. Users also have the option to add/delete roles, add/delete employees,

add/delete departments, update a role, view employees by manager, view employees by department, and access the total utilized salary for each department.
- Technologies used: JavaScript, MySQL, Node.js (Inquirer package, Console.table)
- GitHub: https://github.com/snehitak20/all-of-us | Terminal: use the command “node index.js” to start

**Note Taker (April 2022)**
- An application where the user can write and save notes. Users must first enter both the note title and text before the save button appears. Users can access previous notes from the left-hand column and read their previously-saved notes. Deployed via Heroku.
- Technologies used: JavaScript, Node.js (Express.js Framework)
- Link: https://memory-help-pls.herokuapp.com | GitHub: https://github.com/snehitak20/memory-help-pls

**Team Portfolio Generator (April 2022)**
- A command line application that will generate a new HTML page with the user’s team profiles. Users are prompted first to answer questions about the manager and can choose to answer questions about the engineers or interns on the team. When all questions are finished, a new HTML page is generated with the answers to the questions listed under each member’s profile. Tests are written for this project.
- Technologies used: JavaScript, Node.js (Inquirer package, JEST package)
- GitHub: https://github.com/snehitak20/pick-your-team | Terminal: use the command “node index.js” to start

**README Generator (April 2022)**
- A command line application that will generate a new README to allow users to summarize their projects. Users are prompted within the command line to answer a series of questions that correspond with the appropriate section of the README. When all questions are answered, a new README is generated.
- Technologies used: JavaScript, Node.js (Inquirer package)
- GitHub: https://github.com/snehitak20/read-me-pls | Terminal: use the command “node index.js” to start

**Let’s Stay In (April 2022)**
- An application that is dedicated to creating the perfect Friday night date night in. Users can pick a recipe based on the main ingredient, cook time, and dietary preferences. Entertainment is randomly generated, along with a Techy phrase to impress your date.
- Technologies used: JavaScript/HTML/CSS/Animate.style
- Link: https://snehitak20.github.io/lets-stay-in/ | GitHub: https://github.com/snehitak20/lets-stay-in

**Weather Dashboard (April 2022)**
- An application that can check the current forecast, as well as the future 5-day forecast. Users can search for cities and have their search history be saved to the local storage. Search results can be accessed to check the forecast again.
- Technologies used: JavaScript/HTML/CSS
- Link: https://snehitak20.github.io/chance-of-meatballs/ | GitHub: https://github.com/snehitak20/chance-of-meatballs

**Code Quiz (April 2022)**
- An application that tests the user’s knowledge about basic coding concepts. For each question that is wrong, 5 seconds and 1 point are deducted. Users can log and save their high scores.
- Technologies used: JavaScript/HTML/CSS
- Link: https://snehitak20.github.io/under-the-sea/ | GitHub: https://github.com/snehitak20/under-the-sea

**Work Day Scheduler (April 2022)**

- An application that allows users to enter their events for the day within the time blocks (9am-5pm). Users are able to save the events, and events are stored on page refresh. Time blocks are denoted by color: past (gray), present (red), future (green).
- Technologies used: JavaScript (jQuery/Moment.js)/local Storage/HTML/CSS
- Link: https://snehitak20.github.io/far-too-busy/ | GitHub: https://github.com/snehitak20/far-too-busy