

# Rene A. Cocom, Jr.

1414 W Pratt Blvd Chicago, IL 60626 | 312-890-6166 | [racocom@gmail.com](mailto:racocom@gmail.com)

**Github:** <https://github.com/fasnimo> | **Blog URL:** <https://fasnimo.github.io/> | **Website:** <https://fasnimo.github.io/renecocom/>

---

## FULL-STACK SOFTWARE ENGINEER

---

Software Engineer with an extensive and complementary background in graphic design. Experience collaborating in many facets of the development cycle, including design, UX/UI, and server side-development. Vital customer service and communication skills are built through work in the hospitality industry and as a freelancer—strong passion for creating scalable and reliable software solutions.

---

## SKILLS

---

HTML | CSS

Ruby | SQL | PostgreSQL | Javascript

Ruby on Rails | ReactJS

Adobe Illustrator | Adobe Indesign | Adobe XD | Adobe Photoshop | Adobe Premiere Pro | Adobe LightRoom | Camera Raw

---

## TECHNICAL PROJECTS

---

### Skate Finder

[https://github.com/fasnimo/skater\\_folder](https://github.com/fasnimo/skater_folder)

Collects user's input and returns useful information to the user.

- Utilized React with Rails API MVC.
- Skates Controller action index will render all skates level based on the level column.
- The navigation bar includes routes made with BrowserRouter, and Switch.

### Zodiac Digest

**Client-Side:** <https://github.com/fasnimo/zodiac-frontend> **API:** <https://github.com/fasnimo/Zodica-Plant-API>

Allows for custom inputs to be seen in many cards and modified and created asynchronously.

- Utilized Vanilla JavaScript, Ruby on Rails, MVC, SQL
- The application makes asynchronous requests to the server from the client-side allowed live updates to the DOM.
- I utilized arrow functions and classes.
- Serializers for object relations when requesting associated data.

### Ruby on Rails Application

[https://github.com/fasnimo/my\\_new\\_rails\\_app](https://github.com/fasnimo/my_new_rails_app)

This app utilizes authentication and allows for sessions storing tokens and login and log out for such users.

- Utilized Ruby on Rails, MVC, SQL
- Once a user has logged in, the user can input data.
- Each user can modify their data.
- The application verifies the user's credentials.
- Applications will not allow other users to interfere with other users' inputs.

### Nba Login

**Client-Side:** <https://github.com/fasnimo/nba-react> **API:** <https://github.com/fasnimo/rails-api>

Allows for custom inputs to be seen in many cards and modified and created asynchronously.

- Utilized Vanilla JavaScript, Ruby on Rails, SQL
- The application makes asynchronous requests to the server from the client-side allowed live updates to the DOM.
- I utilized arrow functions and classes.
- Serializers for object relations when requesting associated data.
- Utilizes React Router Dom
- Authentication with Rails API

## EXPERIENCE

---

### **Flatiron**

Chicago, IL

Co-op

June 2019 - October 2020

- Created open-source projects using modern languages, including Javascript, Ruby, and React, while working alongside experienced industry professionals.
- Developed full slack applications with API and Client-Side interface
- Built an Oauth secured database for user queries.

### **Freelance Graphic Designer**

Chicago, IL

Independent

June 2018 - April 2020

- Built a professional-level active portfolio using the Adobe Suite, including Photoshop and Xd for business and nonprofit clients.

## EDUCATION

---

### **Chicago State University**

Chicago, IL

Bachelors in Arts: Graphic Design

August 2014 - May 2016