

# Gabriel Torres

## Backend Software Engineer

thomastgtorres21@gmail.com • 214-714-3840 • [LinkedIn](#) • [Github](#) • [Portfolio](#)

Diligent Backend Engineer skilled in collaboration, Ruby on Rails, and API integrations. Committed to writing high-quality code to produce sustainable web applications. I have over 5 years of experience in customer service and a strong foundation in social sciences.

### TECHNICAL SKILLS

---

Backend:	<b>Ruby, Ruby on Rails, Python, GraphQL</b>
Frontend:	<b>Javascript, React, HTML, CSS, Tailwind, Bootstrap</b>
Deployment & DevOps:	<b>Heroku, Git/GitHub, CircleCI, AWS, VSCode</b>
Databases & ORMs:	<b>PostgreSQL, ActiveRecord, SQL</b>
Testing:	<b>Postman, RSpec, Capybara</b>
Workflow:	<b>Object-Oriented Programming, Object-Oriented Design, Test-Driven Development, Agile, SCRUM</b>

### PROFESSIONAL EXPERIENCE

---

Materials Management Specialist, Texas Health Resources, Denton, TX, 2021-2022

- Ensured the accuracy and security of over **1,000,000** financial transactions while efficiently managing package logistics through ERP system PeopleSoft.
- Successfully handled the processing and distribution of packages and financial confirmation.

Grocery Team Member, Whole Foods Market Uptown, Dallas, TX, 2019-2021

- Assisted with closing procedures to ensure a smooth end to the day's operations.
- Facilitated workstation sanitation and organized displays for optimal presentation.
- Trained and onboarded 5 new team members, ensuring seamless integration into the team and adherence to company standards.

### PROJECTS

---

**[Foodie Brain](#) Tech:** PostgreSQL, RSpec, Ruby, Rails, Circle CI, GraphQL, React, AWS  
A single-page React app that communicates with a Ruby on Rails backend using GraphQL. Data is managed in PostgreSQL, with testing using RSpec. The application utilizes ActiveStorage and ApolloService to display photos, which are stored persistently on AWS. Created in-depth testing automation, resulting in reduced bugs and improved reliability. [Launch](#)

**[Sweater Weather](#) Tech:** Rub, Rails, Postman, PostgreSQL, RSpec, Capybara, APIs  
An app that aggregates weather forecasts from several APIs to provide users with a comprehensive view of the weather in their area of interest. In addition to weather information, the app offers a search feature that returns local books or food places based on the user's query. This application is thoroughly tested with mocked external APIs using VCR and designed error-handling JSON-formatted messages for the frontend. It is also Postman-friendly for testing. Sweater Weather features custom user authentication for login and security. The app exposes and consumes multiple API endpoints, simplifying frontend formatting. This application allows users easy access to weather, local books, and food information, making it a valuable tool for travelers.

**[Spellbinder](#) Tech:** Postman, CircleCi, Ruby, Rails, RSpec, Heroku  
Co-created a fully functional application that leverages Magic The Gathering's API, enabling users to create customized decks. Data is stored in PostgreSQL as the database, with a Rails backend and React frontend communicating through CORS. [Launch](#)

**[Whatsup](#) Tech:** Ruby, Rails, PostgreSQL, RSpec, Capybara, Tailwind, Bootstrap, Heroku  
A web blogger application that provides user authentication for login and allows users to create articles with image

attachments using ActiveStorage. The application is built on PostgreSQL for database management and utilizes Tailwind CSS and Bootstrap for the user interface. This app incorporates Hotwire and Stimulus controllers to enhance user interaction and experience. It serves as a Rails monolith app for users to share their thoughts and ideas. [Launch](#)

**[GTs Fortnite Tracker](#) Tech:** Ruby, Rails, PostgreSQL, RSpec, Capybara, React, Heroku

A video game tracker that integrates with Fortnite's API, enabling users to view the latest in-game items available for purchase and search for player statistics. Built with a React front end and Rails back end, it integrates Fortnite's external API to retrieve data and implements a customized internal API to transmit data to the Front End. [Launch](#)

**[Futbol State Tracker](#) Tech:** Ruby

This is a terminal Ruby application that uses an algorithm to analyze team performance for specific seasons and across seasons. Stat\_tracker generates game, game\_team, and team objects. It then stores those objects in arrays as attributes and calls on them to return statistics from specific methods.

**[Battleship](#) Tech:** Ruby

This is a terminal Ruby game that uses an algorithm to render Battleship, a classic board game where players place one or more ships on a grid board. Players then take turns trying to 'sink' the other player's ships by guessing their coordinates. The game ends when one player's ships are all hit and 'sunk'.

## EDUCATION

---

Turing School of Software and Design

**Accredited Certificate in Backend Engineering**

2023

University of North Texas

**Bachelors of Arts - Anthropology, Certificate in Applied Anthropology**

2021