15152 West Sherman Street
Goodyear AZ, 85338
Matthew@Symbiotic.Love
(912)438-1768

Matthew Ford

Intro:

I am a results-driven professional with a proven track record of personal accountability and success. I am highly detail-oriented and thoroughly enjoy solving complex technical problems. I have a highly varied and unique skill set that allows me to feel comfortable in almost any role in the engineering field. I am masterful in the use of AI systems and testing, and can revolutionize workflows using them. I am self-taught except for extensive coursework on teamtreehouse.com totaling ~6200 points.

Skills:

- **Master:**
  - Prompt Engineering and AI interfacing
- **Expert:**
  - HTML5
  - CSS
  - Understanding of Engineering Architecture
- **Advanced:**
  - React.js
  - Python (Pygame, Tkinter, Django, Flask, NLTK)
  - REST API's
  - System Administration
- **Intermediate:**
  - JavaScript
  - SEO
  - Accessibility Optimization
  - Debugging
  - NPM
  - Liquid
  - Algorithms and Data Structures
- **Basic:**
  - SQL
  - Tailwind CSS
  - Node.js
  - Ruby (Gosu)
  - Rust
  - Electron
  - Agile (Kanban/Scrum)
  - SaaS

Certifications:

My HackerRank profile can be found at [Matthew Ford - fordmatthew501 | HackerRank] and I have obtained all available certifications related to my listed skills.

Projects:

A comprehensive list of all of my projects can be found at SymbioticLove (Symbiotic Love) (github.com)

- **NLP AI Training Script** *(ongoing)* - This Python script uses an NLP model and .csv datasets with the columns "question" and "answer" to train an AI, print the prediction accuracy to the console, and save the trained AI's weights and architecture. There is also a small python script that converts .txt files into .csv files that the training script can use. This will be expanded to take in any common dataset extension. This project is located at: Recipe-Bot-Training/Packaged-NLP-v1.0 at main · SymbioticLove/Recipe-Bot-Training (github.com)

- **React/Flask Boilerplate Template** *(complete)* - This is a template for a web application utilizing a React front-end and a Flask back-end. There are also shortcuts included for setup and server initialization for Windows and Mac/Linux. This can be found at: SymbioticLove/React-Flask-Boilerplate: A boilerplate template with a React front-end and a Flask back-end proxied and connected (complete) (github.com)

- **Our Brand Homepage and Store - React and the Shopify Liquid Framework** *(perpetual)* **-** This is our main website and our store page. Our main site is created with React and our store is created in the Shopify Liquid framework. The related code can be found at SymbioticLove/Homepage: Our brand homepage (ongoing/permanent) (github.com).

- **Mother - A Text Adventure Made with Python** *(ongoing)* **-** I am currently working on a highly complex text adventure in Python's Pygame framework. All of the information (including how to play the alpha!) can be found at SymbioticLove/Mother: Our project to make THE definitive text adventure (ongoing) (github.com).

- **SymbioticBot Lite Chatbot - Static Site** *(complete)* **-** This is a lite chatbot UI created using OpenAI's GPT 3.5 Turbo API. We will be turning this into a React component to integrate with our imageGenerator component and host on our main site. The related code can be found at SymbioticLove/Chat-lite: Our lite chatbot powered by GPT 3.5 turbo (complete) (github.com).

- **SymbioticBot imageGenerator - React Component** *(ongoing)* **-** This is a simple and lightweight image generator component using OpenAI's DALL-E API. This component still requires updates to the "loading" state, but is fully functional. We are offering this component for open and free use at SymbioticLove/React-imageGenerator: Our React imageGenerator component powered by DALL·E (ongoing) (github.com) and will release it publicly when it is finished.

- **SymbioticBot weatherWidget - React Component** *(ongoing)* **-** This is a weather widget React component powered by WeatherAPI. This component is highly functional, providing in-depth weather information at just a few clicks while also being able to reduce to 60x60px and be comfortably located anywhere on the screen. I am looking into packaging this as a desktop app using Electron as well as integrating it into our site using a back-end Python framework to cache API requests and data. The full widget still requires mobile compatibility updates. The information regarding this can be found at SymbioticLove/React-weatherWidget: A weather widget created as a React component using WeatherAPI (ongoing) (github.com)

- **Simple Platformer - Ruby** *(ongoing)* **-** I am also making a simple platformer in Ruby's Gosu framework. This is a side project currently and falls lower on the priority list, but it will be updated and completed over time. The basic outline that has been created so far is available at SymbioticLove/Ruby-Platformer: A simple platformer using Ruby and the Gosu framework (ongoing) (github.com)

- **Custom Email Signatures - HTML** *(complete)* **-** I also create custom HTML email signatures. I sell these in our store. An example code is located here: SymbioticLove/HTML-Email-Signature: A simple HTML email signature (complete) (github.com)
  and hosted here: HTML Email Signature (symbioticlove.github.io). There are a couple of easter eggs included in the JavaScript.

- **Calculators - Python and JavaScript** *(complete)* **-** I created 2 calculators for practice, one with Javascript and one with Python's Tkinter library. These are both complete and functional, though the Python calculator has been left unpackaged. The code for the Python calculator can be found at SymbioticLove/PyCalculator: A simple calculator made entirely with Python's Tkinter library (complete) (github.com) and the code for the JavaScript calculator can be found at SymbioticLove/JsCalculator: A simple calculator created with Javascript (complete) (github.com).

Experience:

My field experience consists of conceptualizing and building our businesses eCommerce platform as well as our community site. I continue to provide administration for this platform daily, as well as continued updates, debugging and optimizations. Many of the projects contained within my portfolio are intended to be integrated into this platform. For instance, the weather widget will be implemented on our main website once I have built a Python backend to cache and update the data on the server side to limit API requests using Django or Flask. If implemented in its current state, it would update on the client side each time someone loaded the site, which would bankrupt us in API calls.