

Kristian Mischke

Software Engineer

| Randallstown, MD | 443-875-3124 | kristianmischke@gmail.com | kkcoder.net |

Problem-solver and thinker with over six years of experience in the industry. I enjoy video game development and am very passionate about utilizing AI and ML to help people with their data. I'm working towards a long-term goal of reforming education by bringing it up to speed with modern technology and learner-centric approaches.

Experience

Associate Software Engineer

August 2021–Present

[Ronday Technologies LLC](#)

Remote

- Adapted quickly to changing requirements and priorities, demonstrating flexibility and a growth mindset in a dynamic startup environment.
- Conducted thorough code reviews and provided constructive feedback to peers, fostering a culture of collective learning and high-quality code standards.
- Pursued R&D in *Generative AI* and within a week, rapidly developed a *GPT-4* prompt 3D sandbox world demo.
- Reduced time to add new backend object types from days to minutes by implementing a generic object data structure stored in *Postgres* and validated by JSON schemas, streamlining the development process.
- Integrated *Mixpanel* analytics for data-driven decision-making, enabling the team to make informed decisions and improve overall application performance.
- Successfully integrated *Unity WebGL* building, automated asset exporting, and hosting addressable assets on AWS, improving time to enter platform providing the user with a seamless experience.
- Managed Dev-Ops build pipeline with *Concourse*, ensuring a unified build process for optimal simplicity and efficiency.
- Demonstrated a deep understanding of end-to-end networking stacks, contributed to distributed realtime backend architecture powered by *SignalR*, *Redis*, and *Postgres*.
- Doubled the number of users capable of being in a space at a given time, by unifying the network paradigm.

Data Science Research Intern

March 2021–July 2021

[RedShred LLC](#)

Catonsville, MD

- Fine-tuned *RoBERTa* for classifying emails with .909 median f1-score, which outperformed the 0.78 of the prior models.
- Annotated naval document dataset with *Label Studio* to train task-specific models for segmenting documents and extracting key actors and events with *NER extraction*.
- Highlighted key connections found in naval reports displaying actors, weather events, and locations in a custom interactive dashboard made with *streamlit*
- Used *MLFlow* to manage experiments and track metrics during model training which improved collaboration with team members.

Junior Programmer

Jan 2018–July 2021

[Mohawk Games LLC](#)

Linthicum, MD

- Integrated *mod.io API* into *Old World*; added support for modding with AssetBundles and for Translation mods.
- Assessed capability of *GANs (Generative Adversarial Networks)* through R&D for generating in-game character portraits.
- Developed a Text Manager class integrating Mohawk's localization system with hierarchical text generation
- Implemented the in-game "Event Browser" tool in *Old World* that allows designers and writers to easily modify and create XML files for in-game events.
- Wrote a *Unity* tool that allows developers to observe the dependency relations of Unity assets.

Education

University of Maryland, Baltimore County

May 2021

Bachelor of Science, Computer Science (Game Development Track) Minor in Applied Linguistics
3.936 GPA Outstanding Senior in Computer Science

Continued Learning

Secure Code Warrior: In-depth OWASP Top 10 Awareness

2022

[CD.Training: TDD & BDD - Design Through Testing](#)

2023

Projects

Recurring Moment

Spring 2021

- Conceptualized, Pitched, and Prototyped original idea during the first 3 weeks of class.
- Acted as Lead Designer and interfaced with the Art & Programming teams at weekly meetings.
- Project management with **SCRUM** development sprints and burndown charts
- Implemented core mechanics and sparse data structures to store time-travel data.

GroupFormer

Spring 2021

- **AGILE** and **GitFlow** frameworks for development sprints
- Developed front-end form for setting up the GroupFormer project using **Django**, **HTML**, and **JQuery**.
- Collaborated with teammates to develop algorithm for scoring participant groupings.
- Integrated Django authentication to secure instructor's forms.

Data with a K

2022-Present

Devs with a K LLC

- Ideated data abstraction layer to handle different kinds of data.
- Architected data pipelining suite in **Python** backed by **Cassandra** and **Redis**.
- Developed receipt processing API that stitches multiple receipt images together using **OCR** to extract receipt items and costs.
- Accelerated knowledge of using **Docker** with Nvidia GPUs and AMD RoCM APUs for **Machine Learning** applications.
- Implemented security standards with password rotation using **Vault** and authentication with **Auth0** JWTs

"Magic Random Gathering"

2019-Present

- Developed AI magic card generators combining text and image **Generative AI**.
- Created modified **BNF** grammar for card generation until switching to **GPT-3** when it was released.
- Card art generation with **VQGAN+CLIP** and later **Stable Diffusion** and **Midjourney**
- Printed hundreds of cards & played dozens of games, including at a local game store!

Schess: A Chess Battle-Royal Variant

Aug 2020-Present

schessgame.com

- Acted as the Lead Programmer during and after a 48-hour game jam with three other friends.
- Responsible for game-logic, and networking code using **Remote Procedure Calls (RPCs)** with **Photon Unity Networking (PUN)** in the **Unity 3D** game engine.

Applying the Cascaded Finite State Grammar Induction Model to Trading Card Game Corpora

Fall 2020

CMSC 473 Intro to NLP Class @ UMBC

- Proposed the original idea for [this final group project](#).
- Implemented-with a group of 3 peers-a **Grammar Induction** algorithm in **Python** from an academic paper that uses a cascaded chunking algorithm with **HMMs**.
- We analyzed model performance using perplexity, and we applied it to Trading Card Games like *Magic: the Gathering*, *Yu-Gi-Oh!* and others.

Linux Chess Kernel Modules

Spring 2020

CMSC 421 Operating Systems class @ UMBC

- Implemented a device module in **C** to store and manage chess game states across multiple file pointers; with the option to play against an AI opponent using the min-max with alpha-beta pruning algorithm.
- Only student out of the three sections of the course to complete all the extra credit and be eligible for the course-wide tournament.

Other Note-Worthy Classes from UMBC

2020

- *Computer Graphics* (Spring 2020) Implemented **ray-tracing** algorithm in **C++**. Used shaders and GLEW and GLSL to push vertices to the render pipeline. We used **Git** version control to track progress.
- *Graphics for Games* (Fall 2020). Navigated the **Unreal Engine C++** source code. Projects focused on implementing graphics algorithms as Blueprints, Plugins, and Engine modifications.

Volunteer Work & Clubs

Volunteer Tutor

2014–2021

Crossroads Homeschool CO-OP

Baltimore, MD

- Managed classes containing about a dozen Middle- or High-School students
- Developed HS Programming curriculum based off the book [Learning Processing](#) by Daniel Shiffman.
- Taught students about variables, program flow, and basic problem-solving using [Scratch](#) by MIT.
- Integrated my [YouTube video tutorials](#) for individualized instruction for the 2018-2020 School Years.
- Utilized the [CodeHS](#) online curriculum for managing and tracking student assignments for a HS Web Design class.

Member & Project Lead Programmer

Fall 2018–Fall 2019

[UMBC Game Developers Club](#)

- Acted as Lead Programmer for [Role Playing Gamble](#), one of the club 2018-2019 games.
- Managed tasks with a group of 2 other programmers throughout the duration of the project, using [Git](#) for versioning & merging and [Unity 3D](#) & [C#](#) technologies for development.