Problem Description

Kids need a fun way to get exposed to writing and storying creating.

Proposed Solution

Create an app that allows a user to interactively create and write a story. The system will allow the user to pick a genre and other details on the type of story they would like to create. Then, the user could start writing a story and at any point ask for the help of an ai trained to the parameters they selected and the ai will generate suggestions for what to write next.

Technical Overview

I will create this system using a multi service architecture. I plan to have three services and one client. The services will be Control, Persist, and AI. The AI service will handle directly interacting with the ai models. These models will be trained to generate text to give an interactive storying writing experience. There will be multiple models train on different genres of stories available. It will be implemented using python. Persist will handle the direct connection to the database layer which will be an SQL db. It will be implemented in go. Control will handle authenticating users, and handle gather forwarding request to and from Persist and AI. It will also use web sockets to allow communication to be asynchronous and bidirectional. This will be implemented in Go. The frontend client will be a very basic web app. The front end will not be the focus of this project.

I will train the AI models using free book data sets and will train the models with GPUs using Google's Collaboratory service. The system will allow users to create accounts. Each user will be able to create and have access to their own books. While a user is writing a book, as stated before, the system will provide autocompletion like features to help them write, and generated suggestions will be flavored based on the parameters they selected for the book.

Milestone List – By Week

1. Jan 6 – Project Proposal
2. Jan 13 – Project Proposal
3. Jan 20 – Control / Persist / AI – Figure out libraries/frameworks and basic database tables
4. Jan 27 – Control / Persist – CRUD users and persist authentication across requests
5. Feb 3 – Control / Persist – Make ability to CRUD books
6. Feb 10 – Control – Work on web socket event loop
7. Feb 17 – Control / Persist– Hook up editing book text to web socket loop
8. Feb 24 – AI – Work on API endpoints with pretrained GPT-2 model
9. Mar 2 – Control / AI – Hook up AI API endpoints to web socket loop
10. Mar 9 – AI – train genre/parameter specific models
11. Mar 16 – Spring Break
12. Mar 23 – Control / Persist / AI – allow books to have genre/parameters
13. Mar 30 – Control / Persist / AI – Ensure correct models are use based on book parameters
14. Apr 6 – Final Presentation
15. Apr 13 – Final Presentation / Project Defense
16. Apr 20 – Project Defense