

Team #018-1

AMAZONE

Get in the zone, Ama-zone!

Team Members:

Nolan Ales
Davis Davalos-DeLosh
Aleksy Huhua
Zhouyang Sheng
James Vogenthaler

Application Name:

Agile Writer

Description:

Agile Writer is a smart word processor using the latest innovations in Natural Language Processing (NLP) AI using technology from OpenAI, the first time a comprehensive solution will incorporate the technology in user-facing technology. Using machine learning, the word processor will provide smart writing assistant features such as whole-paragraph autocomplete, proofreading, style transfer (e.g. automatically converting normal language into legalese, detect non-academic writing, etc.), and potentially more.

Users can expect much greater productivity, as if they had a personal expert assisting them throughout their workflow. From eliminating writer's block, to coming up with prompts, to converting one kind of writing to another, Agile Writer is there for professionals to rely on when their work starts to slow.

Vision Statement:

"To provide the world's creators with productivity-focused workflow."

Version Control:

<https://github.com/CU-CSCI-3308-Fall-2021/lab-2-team-formation-team-018-01>

Development Method:

Scrum + Kanban

Jira Link:

<https://csci-3308-fall21-018-1.atlassian.net/jira/software/projects/A01/boards/1/roadmap>

Communication Plan:

Our team will communicate via Discord for the course of this project. The Discord is set up using different chats separated into: general, meeting plans, off-topic, and brainstorming (although more chats may be added in the future). The first meeting will take place on Tuesday, September 14th in-person, but subsequent meetings will take place via Discord video or Zoom. Most communication will be through the Discord app.

Meeting Plan:

We are meeting every Tuesday at 4pm through Discord. Additionally, there is a standup meeting with the TA every Monday, at 4:30PM through Zoom.

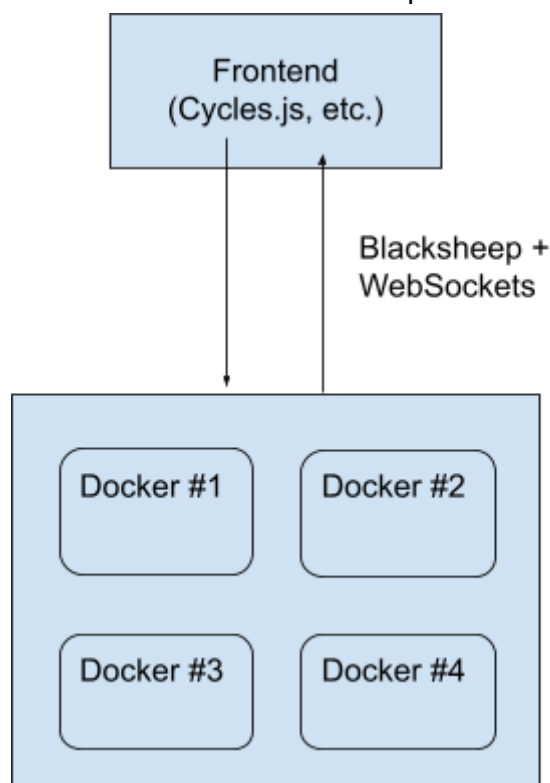
<https://cuboulder.zoom.us/j/91352791900>

Proposed
Architecture Plan:

Backend: Python + Docker

Frontend: HTML/CSS/Javascript - Cycle.js

Communication: Blacksheep framework + WebSockets



Use Case Diagram

