Learning the Popularity Prediction in Information Cascades

Design Document

Team: S23-35
Client and Advisor: Dr. Goce Trajcevski

Ian Johnson - Team Organization
Will Postler - Client Interaction
Paul Brinkmann - Component Design
Evan Gossling - Component Design
Bailey Gorlewski - Testing

sdmay23-35@iastate.edu
https://sdmay23-35.sd.ece.iastate.edu/

Revised: 12/1/2022
Version: 1

Executive Summary

Development Standards & Practices Used

- Scrum Methodology
- IEEE 829 - Software Test Documentation:
- IEEE 830 - Software Requirements Specifications:
- IEEE 1016 - Software Design Descriptions:
- IEEE 12207 - Software life cycle processes:
- IEEE 1028 - Software Reviews and Audits:

Summary of Requirements

Constraints:

- Data parsed under 15 seconds and visualizing outputs of models for graphs made under 15 seconds
- A server to be running with minimum downtime (99.9 % uptime) to host the website and database
- Website must have decent performance (takes less than a minute to load a page)
- Graphs are able to zoom in and out with at most a 5 second delay

Functional/Specification:

- Display graphs/charts representing cascade data
- Display geo-locations of cascade graph data accurately
- Analize / extract from geographical data from author's affiliations
- A website to allow data sets to be inputted and graphs to be outputted

Resource/Physical:

- Visualization tools/frameworks

Aesthetic:

- Website must be designed in a pleasing manner so the webpage is nice to look at and easy to read/understand

User Experiential:

- Website must be easy to use and navigate
- Multiple users need access to the backend

Economic/Market:

- Application and provided services function well on the server provided to us

Environment:

- Application makes efficient use of power on the server and each webpage

Applicable Courses from Iowa State University Curriculum

- COM S 227: Object-Oriented Programming
- COM S 228: Introduction to Data Structures
- COM S 309: Software Development Practices
- COM S 311: Introduction to the Design and Analysis of Algorithms
- SE 329: Software Project Management

New Skills/Knowledge acquired that was not taught in courses

- React
- Mapbox API
- Information Cascades
- Machine learning development

Table of Contents

Dictionary

Information Cascades: When a person makes a decision made solely on the decisions of other people, while ignoring their own personal knowledge to the contrary.

Cascade Graph: A graph that shows you how an initial value is impacted by intermediate values — either positive or negative — and results in a final value. In the context of the project these cascading graphs are generated based on predictive models.

Machine Learning: The use and development of computer systems that are able to learn and adapt without following explicit instructions, by using algorithms and statistical models to analyze and draw conclusions from patterns in data.

Figures and Tables

1 Team

1.1 Team Members

Ian Johnson

Will Postler

Paul Brinkmann

Evan Gossling

Bailey Gorlewski

1.2 Required Skill Sets for Your Project

- Ability to use a Framework with Python to build our website on
- Able to effectively communicate with team members
- Server/database management capabilities with MYSQL
- Ability to make graphs and charts from data and able to integrate them into the website
- Ability to integrate MapBox into our website
- Knowledge of frontend tools (i.e. HTML, JS, and CSS)
- Ability to integrate Machine Learning algorithms for Cascading Prediction Models into our website

1.3 Skill Sets covered by the Team

Ian: Communication, MYSQL, HTML

Will: Communication, MYSQL, Frontend tools, Graphs/Charts

Evan: Communication, Framework tools, Graphs/Charts, Frontend, Python

Paul: Communication, MYSQL, Frontend tools

Bailey: Communication, MYSQL, HTML, Python, Graphs/Charts

1.4 Project Management Style Adopted by the team

Our team will be utilizing the Scrum/Agile Methodology, this is because the scrum methodology has regular team meetings and prioritizes communication. Our team will be working on components that are highly coupled together and will require frequent and effective team communication. The scrum methodology is also more robust and will allow teammates to communicate issues and setbacks encountered during development.

1.5 Initial Project Management Roles

Ian Johnson - Team Organization

Will Postler - Client Interaction

Paul Brinkmann - Component Design

Evan Gossling - Component Design

Bailey Gorlewski - Testing

2 Introduction

2.1 Problem Statement

In the current digital age the Internet, social platforms such as Twitter, Weibo, WeChat, and more have become the main source of information in people's daily lives. Various news, events, and posts are spread through social networks. Similarly, information is spread around the academic world through papers. These papers get read, shared, and cited, similarly to how tweets get liked, retweeted, and quoted. Therefore, predicting the effect of an individual paper after a certain time period (in terms of some of the metrics previously mentioned) has garnered the attention of academics and publishers. Measuring and predicting the propagation of papers shows how much of an impact each one has had / may have, which is good for both evaluating a paper's quality and relevancy (e.g. if it's 20 years old and never been cited then it's likely irrelevant or poorly executed).

For example, the American Physical Society (APS) contains scientific papers published by APS journals (https://journals.aps.org/datasets). Every paper in the APS dataset and its citations form a citing cascade. Many researchers work on addressing the problem of information cascades using this APS data set. However, in the past, developed models (e.g. deep learning techniques) have not payed attention to:

1. The users' need to look up data in the APS or similar data set
2. Visualizing citation data on a map
3. Handling updates to the dataset and informing users

The objective of this project is to develop an end-to-end system that will provide such functionalities for the users interested in visualizing locations related to popularity cascades in the context of scientific paper citations.

2.2 Intended Users and Uses

There are a few broad categories of users that would benefit from such systems:

1. Individual scientists can see what papers are likely to gain citations, or look for prolific papers and predictions for related papers
    a. See which papers have large amounts of citations.
    b. See visualizations of data across the country based on these citations
    c. View trends over time of citation usage across country
    d. See how their own work stands against other researchers.
    e. Look for more prolific research topics.
2. Funding institutions can see which universities are likely to gain recognition due to the increase of citations and reasons about their geographical distributions
    a. See which papers have large amounts of citations.
    b. See visualizations of data across the country based on these citations
    c. View trends over time of citation usage across country
3. University administrations that want to compare scientific impact of its faculty personal in with peer institutions based on geographic whereabouts
    a. See which papers have large amounts of citations.
    b. See visualizations of data across the country based on these citations

c. View trends over time of citation usage across country
d. Provide funding based on data

Personas:

Individual Scientist - Sam

Sam is a PhD researcher working towards tenure at their University. Sam wants to make sure that they are researching topics and publishing papers that will be cited frequently to increase their notoriety within their field of academia.

Funding Institution - Frankie

Frankie is a staff member at an institution that provides funding for research projects. Frankie needs to be able to see papers with large amounts of citations, to see which topics their institutions can provide funding for in the future. This will allow Frankie to successfully fund research that will have a larger impact on academia/overall world research.

University Administrator - Alex

Alex is an administrator who enjoys promoting their institution. Alex likes to back up their praise with details about how their institution has written research papers that have been cited across the country/world. This data could allow Alex to recruit and retain researchers at their institution.

2.3 Requirements & Constraints

Functional/Specification:

- Display graphs/charts representing cascade data
- Display geo-locations of cascade graph data accurately
- Analize / extract from geographical data from author's affiliations
- A website to allow data sets to be inputted and graphs to be outputted
- Data parsed under 15 seconds (constraint) and visualizing outputs of models for graphs made under 15 seconds (constraint)

Resource/Physical:

- A server to be running with minimum downtime (99.9 % uptime (constraint)) to host the website and database
- Visualization tools/frameworks

Aesthetic:

- Website must be designed in a pleasing manner so the webpage is nice to look at and easy to read/understand

User Experiential:

- Website must be easy to use and navigate
- Multiple users need access to the backend
- Website must have decent performance (takes less than a minute to load a page - constraint)
- Graphs are able to zoom in and out with at most a 5 second delay (constraint)

Economic/Market:

- Application and provided services function well on the server provided to us

Environment:

- Application makes efficient use of power on the server and each webpage

2.4 Engineering Standards

IEEE 829 - Software Test Documentation:

Justification: It is necessary to write proper documentation and test cases for our project. This documentation will allow our code to be understood by others, and potentially be expanded upon in the future. Proper testing allows us to achieve a high level of coverage on the code that is written, and provides proof of specified functionality.

IEEE 830 - Software Requirements Specifications:

Justification: It is essential that we have set software requirement specifications, in order to be clear in the scope of our project, and to have set milestones for what must be accomplished.

IEEE 1016 - Software Design Descriptions:

Justification: We are fulfilling this standard already by having a concrete set of design documents, it is important that our full design process is documented.

IEEE 12207 - Software life cycle processes:

Justification: For this project we will be utilizing an agile software development life cycle, this will allow us to continuously improve/test our software. Having a set SDLC process, will allow us to be clear in how we will go forward in our development.

IEEE 1028 - Software Reviews and Audits:

Justification: Reviewing the code written by each team member will allow us to catch bugs that may not be otherwise found, and provides a 2nd set of eyes on what has been contributed. Code reviews are an essential part of the SDLC, and are used in most enterprise software projects.

3 Project Plan

3.1 Project Management/Tracking Procedures

Our project will be adopting an agile methodology for our project management. The main goal of this project is to evaluate a kernel-based structural-temporal cascade learning model to explicitly estimate and encode the structural similarity of cascades with graph kernels. To accomplish this we decided it would be best to use the agile methodology because we need to constantly test and update our software while developing it. We will also have cross functional teams that are working on multiple segments at the same time. Since these segments are closely related, agile is preferred as it allows the needs for each segment to constantly develop.

We will use GitLab to track our team's progress. GitLab has Boards, Service Desk, and Milestones each of which will allow our team to communicate progress with each other. This will improve overall project structure as each member can stay up-to-date on current project tasks. We also use Discord to communicate, as it is a convenient way to easily communicate online.

3.2 Task Decomposition

Task	Title	Description
0	Design Infrastructure	Formalize the frameworks and application structures.
1	Create Local Environment	A local environment implementing the structure from task 0 is created and shared among developers.
2	Initialize Database	An SQL database is initialized with given data.
3	Implement Algorithms	Make sure algorithms are implemented and produce values that are expected based on algorithms given to us.
4	Design UI	Design a web frontend that allows user selection for proximity and diversity at different weights for data attributes.
5	Implement UI Design	Implement and develop the UI as designed in task 3.
6	Create Queries	Have all necessary queries for each function finalized.
7	Implement backend ML logic	All algorithms for parsing and representing data are implemented in the backend.
8	Add Backend REST Endpoints	Endpoints to access data from task 6 from the frontend are available.
9	Add UI functionality	Add different assorted quality of life functionality to the UI.
10	Create Production Environment	Add data visualization tools to the frontend.

	*Data Visualization	
11	Help documentation	Include documentation to help users understand what they are looking at and troubleshoot problems they run into using the application.
12	Presentation	Give a presentation about the project.

Table 3.1 Task Decomposition

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

- Milestone 1: Architecture Design (Oct. 28)
    - We will have a diagram that shows how the frontend and backend will work together. Our diagram will be complete, but also flexible to account for any changes that need to be made in the future.
- Milestone 2: Finalize Algorithm Solutions (Nov. 11)
    - By this date, we will choose algorithms we will use to determine how to generate the population cascades for the dataset.
- Milestone 3: Finalize Design Document (Nov. 18)
    - The design document will be comprehensive and contain all information necessary for the development of the software. The document will contain all design plans for the software, how the team will work together to develop the software, and the plans for testing the quality of the software.
- Milestone 4: Revising Design Selections and Criteria (Dec. 2)
    - We will have revised and finalized the requirements and criteria for success to be as accurate as possible before we venture into the second semester.
- Milestone 5: Complete Unit Testing (Dec. 9)
    - We will implement test cases for each component of the application that will be used to ensure proper output during development. Each component will be tested for vulnerabilities, things that will cause the server to crash, and general correctness.
- Milestone 6: Finalize Presentation (Dec. 9)
    - We will ensure our presentation for the faculty judges has accurate information and enough information. We will go through a few practice runs as well to ensure each member knows what they will be saying and which ideas they will be communicating.

- Milestone 7: Complete Integration Testing (Jan. 27)
    - We will test each component for compatibility and vulnerabilities. We will also test component interactions for any vulnerabilities or problems that could be caused.
- Milestone 8: Alpha Release (Mar. 33)
    - A simple web application that displays the information from at least one dataset ran through at least one learnt algorithm. Show the map and provided graphs, and have decent descriptions for each graph.
- Milestone 9: Beta Release (Mar. 24)
    - The application should meet at least 90% of the design requirements outlined in the Design Document. Should allow users to select different datasets and different algorithms. All functionality should be there with minimal bugs.
- Milestone 10: Final Software Version (Apr. 21)
    - The final version should be finished. UI will be finalized, bugs ironed out, functionality finalized, and design requirements met.
- Milestone 11: Final Presentation (May 5)
    - The final version of the presentation will include a demonstration of the completed project highlighting different features of the software.

3.4 Project Timeline/Schedule

CPRE 491	10/14	10/21	10/28	11/4	11/11	11/18	11/25	12/2	12/9
Task	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14	Week 15	Week 16
Design Infrastructure									
Design UI									
Create Local Environment									
Presentation									

CPRE 492	1/20	1/27	2/3	2/10	2/17	2/24	3/3	3/10	3/17	3/24	3/31	4/7	4/14	4/21	4/28	5/5
Task	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14	Week 15	Week 16
Initialize Database																
Implement Algorithms																
Implement UI Design																
Create Queries																
Implement backend ML logic																
Add Backend REST Endpoints																
Add UI functionality																
Data Visualization																
Help documentation																
Presentation																

Table 3.2 Project Gantt Chart

3.5 Risks And Risk Management/Mitigation

For each of our tasks, the risk probabilities are pretty low. We have experience with most of these tasks and therefore know how to implement them.

Task	Title	% Risk	Reason	Mitigation Strategy
0	Design Infrastructure	0.1	Planning Stage	N/A
1	Create Local Environment	0.2	Initialization Stage	N/A
2	Initialize Database	0.4	Initialization Stage	N/A
3	Implement Algorithms	0.5	Initialization Stage	Make sure algorithms are implemented and produce values that are expected based on algorithms given to us.
4	Design UI	0.3	Planning Stage	N/A
5	Implement UI Design	0.4	Initialization Stage	N/A
6	Create QueriesQueries	0.7	Performance Risk	Make sure that queries meet performance levels. Considering that there is a large amount of data, timely queries are very important.
7	Implement backend ML logic	0.9	Performance Risk	Algorithms need to be thoroughly tested to make sure they meet performance criteria.
8	Add Backend REST endpointsEndpoints	0.5	Tool Failure Risk	Issues with REST tools may arise. Other tools may need to be looked into for backup options.
9	Add UI functionality	0.4	Performance Risk	N/A
10	Create Production Environment *Data Visualization	0.5	Performance Risk	Possible issues with performance. Mitigation may be different tools, increased testing, and possibly no implementation.
11	Help documentation	0.1	Documentation	N/A
12	Presentation	0.1	Documentation	N/A

Table 3.3 Risks and Risk Management

3.6 Personnel Effort Requirements

The following Table 3.4 outlines the total person-hours estimated to complete each task in the project.

Task	Title	Hours	Explanation
1	Design Infrastructure	90	The frameworks and application structure form the foundation on which the application is built.
2	Create Local Environment	90	A local environment is necessary to develop in and verify proper functionally of application before deployment.
3	Initialize Database	40	SQL databases with the given data will be utilized for proximity and diversity algorithms in the backend.
4	Implement Algorithms	40	We will have to take existing algorithms, figure out how they work, be able to programmatically and flexibly send data to them, and analyze their output and turn that into data we can display.
5	Design UI	60	Create visual design for the frontend UI.
6	Implement UI Design	40	The UI forms a large portion of this project. This is the medium in which the users of the app will interact with the data.
7	Create Queries	60	The queries used with the SQL database will be somewhat complex to account for proximity and diversity weighting.
8	Implement backend ML logic	80	The backend will need to take the algorithms implemented above, and be able to utilize them based on the data inputted.
9	Add Backend REST Endpoints	60	This simple task will make accessing the data from the frontend much easier.
10	Add UI functionality	80	Fully functional ability of frontend to communicate with backend and data requests.
11	Data Visualization	80	The app is now functional and can be deployed to the public user. * If time permits, utilization of data visualization techniques in frontend would add a nice feature to the app.
12	Help documentation	40	Documentation will need to be created to comment our code, as well as explain how to use our web application.
13	Presentation	110	A final presentation will need to be created in order to showcase the functionalities of our project, as well as the design process.

Table 3.4 Personnel Effort

3.7 Other Resource Requirements

For additional resources, we need a server to manage the database and the machine learning algorithms + external training databases from our advisor. Besides that, there is no other resources needed besides open source software we will be utilizing (APIs like the MapBox API).

4 Design

4.1 Design Context

4.1.1 Broader Context

One of the main objectives of information cascade popularity prediction is to forecast the future size of a cascade given the observed propagation information. It is an enabling step for many practical applications (e.g., advertisement, academic writing, etc.). Recent advances in neural networks have spurred a few deep learning-based cascade models, which preserve the structural features of information cascades with node embedding and graph neural networks. However, most of the efforts in cascade graph learning as well as its internal temporal dependency, mainly focus on node-level similarity learning, ignoring the structural equivalence among different sub-graphs that are more informative for information diffusion prediction. The main goal of this project is to evaluate a kernel-based structural-temporal cascade learning model to explicitly estimate and encode the structural similarity of cascades with the graph kernels. We will also employ a non sequential process to address the temporal dependency which, in turn, can be used to facilitate information popularity prediction. The methodology will be evaluated on real datasets, for which a (front-end) visualization is one of the objectives of this project.

Area	Description	Examples
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	Increasing/reducing exposure to pollutants and other harmful substances, increasing/reducing safety risks, increasing/reducing job opportunities
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	Development or operation of the solution would violate a profession's code of ethics, implementation of the solution would require an undesired change in community practices
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	Increasing/decreasing energy usage from nonrenewable sources, increasing/decreasing usage/production of non-recyclable materials
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups.	Product needs to remain affordable for target users, product creates or diminishes opportunities for economic advancement, high development cost creates risk for organization

Table 4.1 Broader Context Considerations

4.1.2 Prior Work/Solutions

To our knowledge, there is no visualization tool that can provide predicted mappings of the popularity of scientific works. Some agencies like NSF have visual analytics but only for existing projects, not for predicted ones. While there exists certain works addressing prediction models we will focus on relatively recent results that have developed ML models for predicting the scientific impacts. Our two main sources for the algorithms and data are from these two papers, which detail using algorithms based on Recurrent Cascades Convolutional Networks (CasCN) (Information Diffusion Prediction via Recurrent Cascades Convolution) and Variational Cascades (VaCas) (Variational Information Diffusion for Probabilistic Cascades Prediction) to predict cascades.

- X. Chen, F. Zhou, K. Zhang, G. Trajcevski, T. Zhong and F. Zhang, "Information Diffusion Prediction via Recurrent Cascades Convolution," 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2019, pp. 770-781, doi: 10.1109/ICDE.2019.00074.
    - https://ieeexplore.ieee.org/document/8731564
- F. Zhou, X. Xu, K. Zhang, G. Trajcevski and T. Zhong, "Variational Information Diffusion for Probabilistic Cascades Prediction," IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, 2020, pp. 1618-1627, doi: 10.1109/INFOCOM41043.2020.9155349.
    - https://ieeexplore.ieee.org/document/9155349

4.1.3 Technical Complexity

The architecture of the system will have to provide a frontend (UI), with a backend that will do most of the processing, data storage, etc. In addition, different components will have distinct subsystems that will have to be integrated. For example, the UI will have to combine map display, with the various menu options. The design will also have to consider the execution of multiple algorithmic solutions, as well as integrating the heterogeneous data. The UI will also utilize a mapping API such as Google Maps or Mapbox to show the actual map data.

The challenges of this project involve coupling of outputs of cascade graph prediction algorithms with displaying geospatial values of attributes that are secondary to the ML solution. This will allow users to see geospatial visualizations of the prediction algorithms, and make decisions based on them.

4.2 Design Exploration

4.2.1 Design Decisions

It has been noted that Google Maps has issues with its API. It has more limitations (with the free version) and also has less performance than MapBox. It is also less intuitive to use. Therefore, we plan on using Map Box as it is more intuitive, especially for preliminary projects.

ESRI is the most comprehensive industry standard for geographical data and mapping. However the data that we will be inputting into ESRI will not completely comply with ESRI's input standards. This may cause overhead as we will need to make the data match ESRI's format.

An important design decision we will have to chose is the data storage management we will use, straightforward relational (e.g., mySQL) to public available spatial databases (e.g., QGIS)

We are currently deciding on which middle-ware technologies we need to select for effective generation of objects between frontend and backend (JSON, XML etc.)

4.2.2 Ideation

To generate ideas for our design decisions we created a Lotus Blossom.

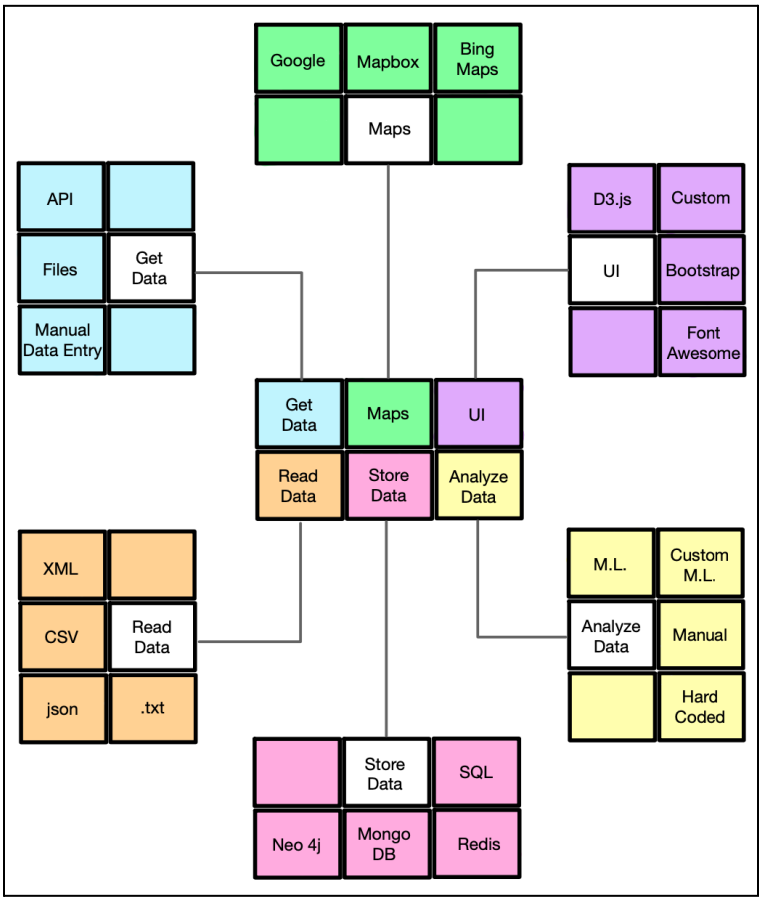

Figure 4.1 Ideation Lotus Blossom

For mapping, we had a few options to consider. One of the first choices we eliminated was Fencer, an API that we had heard of, but was discontinued in 2018. That limited us to one of three choices: Mapbox, Google maps, and Bing maps. We looked at a variety of factors on each of these different APIs (decision matrix below), and ended up picking Mapbox.

For our "Get Data", the design choices were pretty simple. It does not make sense to choose "manual data entry" for our website, so we can immediately eliminate that. Files and API are both good choices. We can use the Files option for users, as this will enable them to upload their own dataset to generate the predictive models. Using an API will be a good option for when we want to pass the backend data to the frontend for our charts, maps, and graphs.

For our "Store Data" part of the Lotus Blossom, we have considered using a SQL database, Redis, MongoDB, or Neo4j. Each of these database options provide their own strengths and weaknesses, so integration testing will be needed to see which one will perform the best.

4.2.3 Decision-Making and Trade-Off

After our ideation process to come up with multiple map API options, we used the following decision matrix to finalize our decision (Table 4.2).

	Selection Criteria	Weight	Google Maps	Mapbox	Bing Maps
1	Documentation	0.4	9	9	6
2	Ease of use	0.3	7	7	6
3	Performance	0.2	8	9	6
4	Afordability	0.05	4	7	10
5	Aesthetic	0.05	7	8	3
6	Total	1	35	40	31

Table 4.2 Map API Decision Matrix

As a team we created several considerations and criteria for the map API we would use including available documentation and performance. Then we researched each map API on each criterion. We then chose Mapbox for our project, because it had the highest weighted score in our decision matrix (Table 4.2). Mapbox has similar documentation and tutorials to Google Maps, however, Mapbox has a better performance and price structure for our project. Overall, Mapbox will be best suited for our project's needs.

4.3 Proposed Design

4.3.1 Overview

Our design is, in the most general sense, a website. It will consist of the following five pages: a homepage, an about us page, a model creation page, a prediction page, and a view page. The homepage will contain a brief tutorial / explanation of how to use our tool, and the about us page is pretty self explanatory. The main functionality we are providing, as described in later sections, is provided by the remaining three pages. Each of these works towards providing the core functionality we are trying to achieve, and will allow for things like the creation of Machine Learning models and displaying of predictions (location and otherwise) of information cascades.

Current version for the design is targeting the main components of a system that will enable the desired behavior. To better illustrate the subsequent sections of this document we provide a use case illustration below:

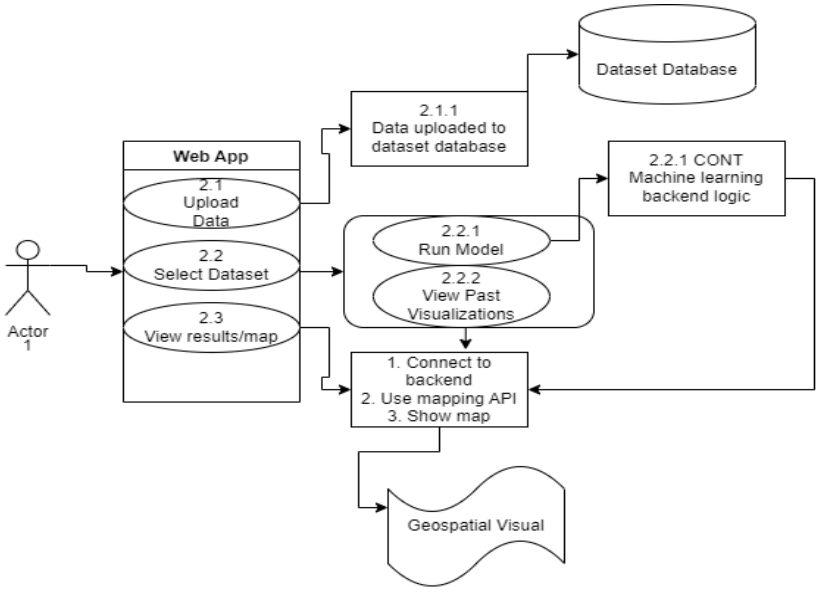

Figure 4.2 Use Case Diagram

4.3.2 Detailed Design and Visual(s)

Based on our use case model (cf. Figure 4.2), here is our diagram of our system and subsystems:

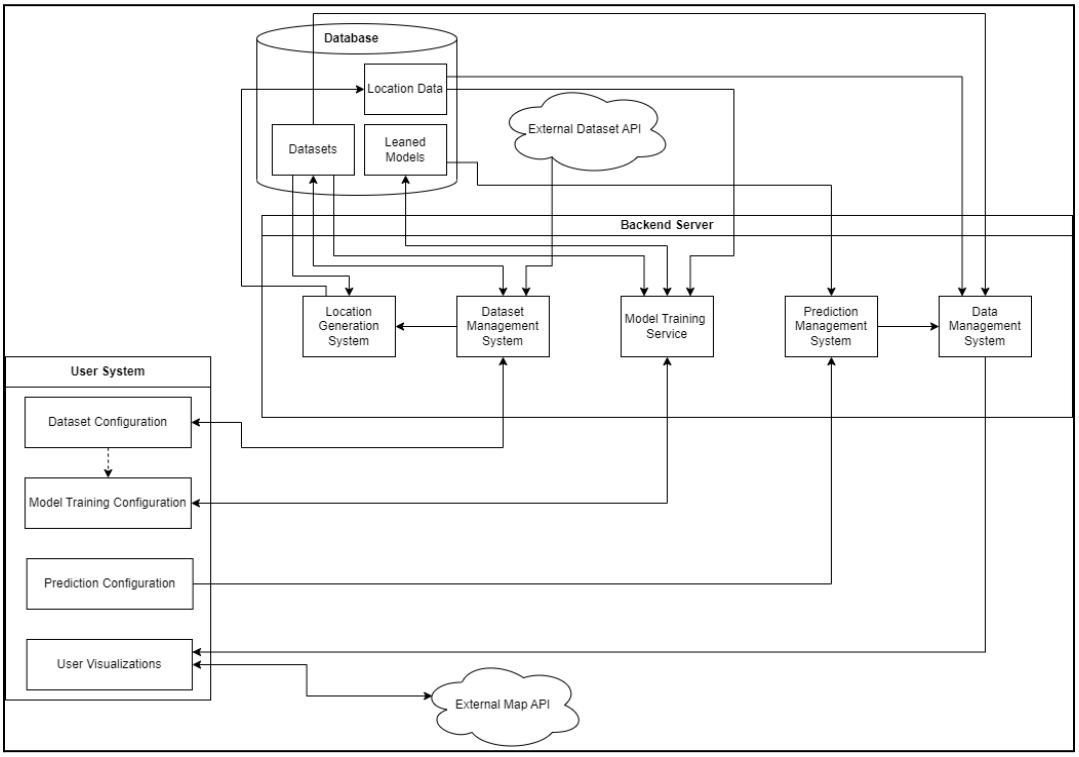

Figure 4.3 System Block Diagram

Now we describe each subsystem and their interactions:

Dataset Configuration:

This subsystem is on the user's system and is responsible for gathering the user's selected dataset and communicating with the backend dataset management system to load the dataset into the database. The dataset configuration will then move the information to the model training configuration.

Model Training Configuration:

This subsystem is on the user's system and is responsible for gathering the user's training model selection and parameters. This subsystem is responsible for sending this data to the backend so that it can be processed by the Model Training logic.

Prediction Configuration:

This subsystem is on the user's system and is responsible for gathering the user's prediction settings and parameters. This module communicates this selection to the backend prediction logic.

User Visualizations:

This subsystem is on the user's system and is responsible for visualizing the predictions generated from the prediction model. It will display graphs, maps, as well as some of the settings users can change about what they are viewing.

Location Generation System:

This subsystem is on the backend server and is responsible for using the dataset to compute the location information. This component also stores that information on the database for later processing.

Dataset Management System:

This subsystem is on the backend server and is responsible for managing the datasets. This module gets the dataset selection and loads a dataset from an external API or from the user. This component is also responsible for translating these datasets to a format understood by other subsystems and stores it on the database.

Model Training System:

This subsystem is on the backend server and is responsible for creating the learnt models to be read by the Prediction Management System. This module utilizes the training algorithms and user's training parameters. It also interacts with the database to read the datasets and location data, as well as, storing the learnt model.

Prediction Management System:

This subsystem is on the backend server and is responsible for taking in the user's prediction settings and creating a prediction on the learnt model stored in the database.

Data Management System:

This system interfaces directly with the database and Prediction Management System to give the user the data for visualization that they need.

4.3.3 Functionality

Users are intended to do three main things. Firstly, they can upload custom datasets to create Machine Learning models, which they can then use to predict the popularity cascade of their paper proposal. Secondly, users can use a premade predictive model for the predictions, and finally they can look at the popularity cascades of papers that have already been written. The information shown about the (predicted or viewed) cascades will include graphs displaying information such as number of cascades over time, the number and size of cascades, and the location data of the cascades. To put it more simply, the number of cascades, number of citations, and where the citations are. Here's a prototype of the prediction page:

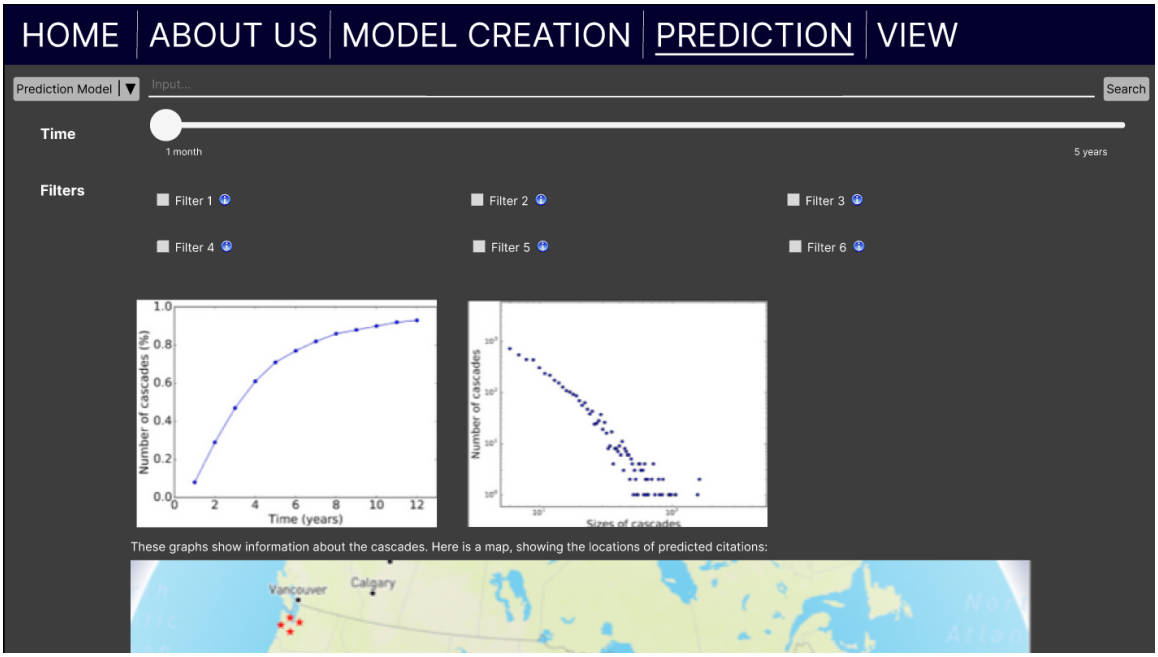

Figure 4.4 Prediction Page Prototype

On this page, users can select a prediction model, filters, and how far out the prediction goes. Below that is the view component, where users can visually see the information described above. As well, here is a prototype of the model creation page:

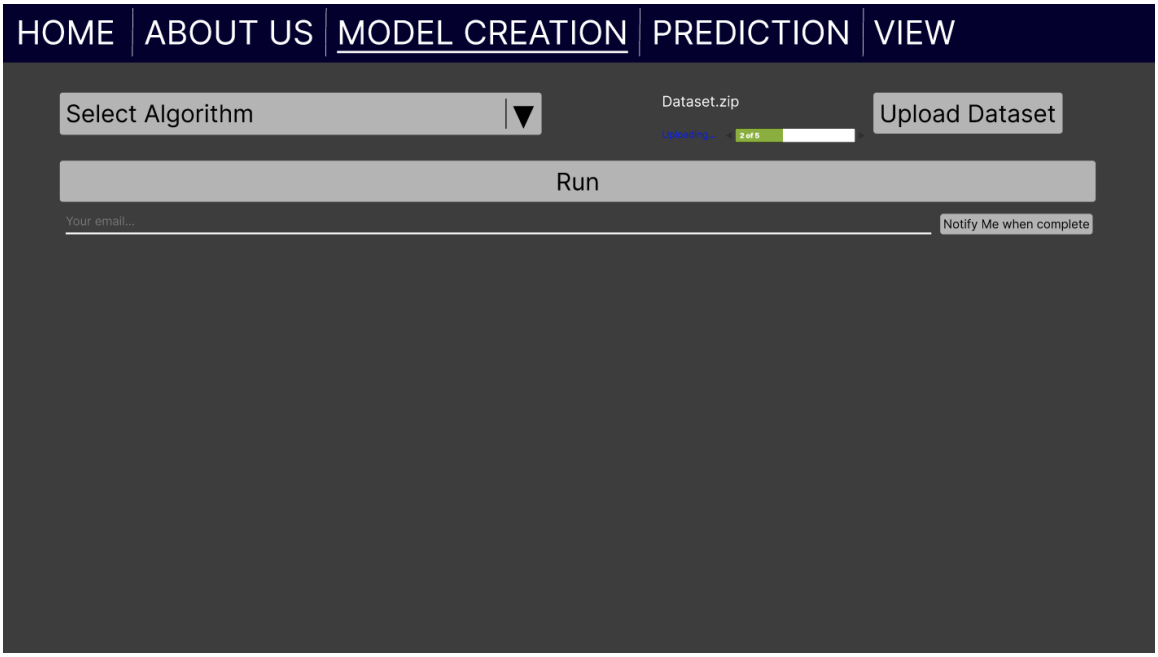

Figure 4.5 Model Creation Page Prototype

Here, users can select which algorithm they wish to use on their dataset, upload their dataset, and input their email address so they can be notified when their prediction model is done being developed.

4.3.4 Areas of Concern and Development

We are having constant discussions with our client to ensure that all our requirements meet their needs. We still have a few design decisions to make, but we've established criteria for each one that will help us ensure each decision meets the needs. For example, one of the biggest needs expressed was to be able to visualize geo-location data of citations on a map. We detailed that decision in Section 4.2.3 of this document. Just like with that decision, we will be / have been thorough in the creation of evaluation criteria for each decision, allowing us to act confidently in our decisions. We will also be going over our decisions and why we made them with our client to ensure they feel that their needs are met.

While we were considering the different frameworks and technologies to use (cf. Section 4.4), notwithstanding their benefits for the implementation, it turned out that we need to gain more experience and hands-on practice with some of them. Our primary concern for the time being is becoming proficient with them and integrating the sub-components of the overall systems. We will continue to communicate with our client in order to make sure that our product design meets their requirements.

The immediate plan is to allocate tasks corresponding to getting familiarized with technologies, as well as portions of time among the team members who will be in charge of their corresponding role in the sub-components. We will need discussions related to integration testing scenarios.

4.4 Technology Considerations

MySQL: Most of the team already knows how to use it and it is pretty standard within the industry. We need to store data and this will be the best way to do it.

MapBox: Mapbox has similar documentation and tutorials to Google Maps (competing software), however, Mapbox has a better performance and price structure for our project.

React/Flask: These are very popular frameworks and they are easy to set up and use. There is also an abundance of tutorials online for them so we can easily learn how to use them. A team member already knows how to use Flask as well, therefore it makes sense to use.

4.5 Design Analysis

So far, the main things we have done have already been covered in this document. Making the figma prototype, selecting which libraries and languages we will be using, planning out how all the smaller components will interact, and our project schedule. Our project has been going well, and our client has approved of our progress.

5 Testing

In this section we present the details of our plan to conduct various tests throughout the development of the project. We note that the functional and nonfunctional requirements were described in Section 1.1 and they were translated into respective system components in Section 4.3. We will refer to Figure 4.2 and Figure 4.3 of Section 4 when describing the envisioned testings.

We note that cost related issues are not addressed since the objective of this project is to provide a proof of concept implementation, not a full fledged product. We also note that while following the good practices for testing, our plan is adapted to the specific needs of the project. In that regard, it is comprehensive with respect to the requirements provided by the client.

5.1 Unit Testing

The first group of unit testing corresponds to different components of the UI. These components include: the Dataset Configuration, Model Testing Configuration, Prediction Configuration, and User Visualization systems. The second group of unit testing corresponds to the backend components. These components include: the Location Generation System, Dataset Management System, Model Training service, Prediction Management System, and Data Management System.

For each of these systems we will check: whether the input is appropriate and the accuracy of the component output. For example, The Dataset Configuration tests will need to check whether the component can properly identify valid datasets and reformat them for use in the database. The User Visualization component will be tested on if the component can accurately display data to the user and can be interacted with. The Model Training Service will be tested to see if the component can accurately train models used for the predictions.

Since different team members will work on different units, and moreover different stages and their different milestones, certain units will be tested early on (eg. displaying maps). Each group member will be describing (and, where applicable, implement) unit tests as each part is started.

The unit testing tools depend on the software utilized in each component. For Example, the UI components will be written in Javascript, so we will be making unit tests using Jest. For components in the UI that require dependencies, we will be using the React Testing Library. This library is used to create fake instances of required dependencies in order to test the unit.

5.2 Interface Testing

To see interfaces in our design please Refer to Figure 4.4 and Figure 4.5, we have multiple dropdowns in our design: Citation Database, Model, Filter, Data Upload, and Time. We will have basic case values (ex. which prediction model to follow). We will need to interface outcomes of the algorithms and map them to the UI, those are the settings we will be testing.

We have many interfaces in our design. The biggest one we will need to test is between the front-end and back-end. The most important integration path between those two is the integration between the Prediction Configuration module (front-end) and the Prediction management module (back-end), which then goes to the Data Management system (back-end), which then goes to the User Visualization module (front-end), which then also needs to integrate with the External Map API (Likely Mapbox). There are a lot of moving parts in that chain and we'll need to test each one.

For example, we will need to ensure that the Prediction Configuration module will be able to properly pass off the configuration input by the user to the Prediction Management module (and, since it's user inputted data, that getting wrong / malicious data does not take down the whole system). Testing to ensure that the configuration entered matches the configuration being received will be very important.

5.3 Integration Testing

Unit testing and Interface testing are both critical and will need to be done before we move onto Integration testing. There are several critical paths, forward path (front UI to backend) and backward path (backend to UI). We will proceed with full tests in terms of appropriately identifying user requests, and correctly checking the answers and the display/locations.

Some tests will be conducted earlier in the development stage (see Gantt chart in Section 2.4). For example, we planned that by mid-February we will be able to conduct the forward integration tests in parallel. We will conduct the integration testing in terms of the backend components that are supposed to select dataset, algorithms, etc. The backward path from backend to UI is expected to be completed in late March or early April and it will be the last stage of the system testing.

5.4 System Testing

We will draft scenarios that will include all of the testings described in the previous Sections (5.1 - 5.4) in a seamless scenario. For example, we will have a full test that will involve user selecting citation database and prediction models. If the database is not available or the model has not been trained, the system will provide a corresponding error message and will ask the user to either change selection or wait until the datasets' respective training algorithms have been obtained. Conversely, if both database and model are available, we will generate the predictions in terms of most cited papers and display the respective institutions with which the authors are affiliated.

5.5 Regression Testing

Once the system testing has been completed in a satisfactory manner. We will try to augment the citation databases (i.e., add one more database), as well as add additional models. We will test how these extensions affect the UI components and the different interfaces. Our connection to the mapping API is a critical feature that we need to work on every test as the geospatial visualization is one of the main features of the project. Connection to the citation database is also an essential part of the project. Just like in the unit testing, Jest and the React Testing Library will be the tools used for regression testing.

We expect that these tests will be done by mid April.

5.6 Acceptance Testing

When it comes to functional and nonfunctional requirements being met, we will directly involve the client. The acceptance of the product will be decided based on criteria such as: intuitiveness of the UI, efficiency of the overall response time, effectiveness in terms of the accuracy of the model. Also, we will involve the client in the different stages of integration testing too. Based on feedback from the client, more testing will be added, or the final design will be accepted.

5.7 Security Testing

While we recognize the importance of security and privacy requirements, these are not the objectives of this project. However, we will provide basic login functionalities and checking if a user is registered. Passwords will be hashed and we will need to test that they are able to provide security as well. Besides this, there is not much security testing that is necessary for our project.

5.8 Results

Testing has not yet been completed.

6 Implementation

For next semester, the immediate goals of the implementation of the project are to develop a working database to store users, and potentially their datasets. Additionally, we will develop a web application to implement data upload and data display (graphs/charts/maps). The first tasks to be completed include initializing the database and framework. We will then go into implementing the algorithms and creating queries. This will then continue into adding backend machine learning logic and REST endpoints. Finishing up into adding full UI functionality and data visualization.

7 Professional Responsibility

This discussion is with respect to the paper titled "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", International Journal of Engineering Education Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

After reviewing "Contextual Professionalism in Capstone Projects Using IDEALS Professional Responsibility Assessment" by McCormack et al. we could identify the seven areas of responsibility found in Table 1. This table offers nice qualitative observations for each of the areas and how they map to the NSPE canons. Below is a continuation of the table where we analyze each area to their best fitting societal code of ethics, their impact on my team's project, and my team's current performance.

Area of Responsibility	**Best-Fit Society Ethics Code**	**Importance**	**Our team's Performance**
Work Competence	ACM 2.1, 2.2, and 2.6	High	Medium
Financial Responsibility	IEEE CS 3.09	Low	Medium
Communication Honesty	IEEE COE i.3-i.5	High	High
Health, Safety, Well-Being	IEEE COE i.1	Medium	N/A
Property Ownership	ACM 1.6 and 1.7	Low	N/A
Sustainability	IEEE COE i.1	Medium	Low
Social Responsibility	ACM 3.1	High	High

Table 7.1. The Seven Areas of Professional Responsibility

After reviewing the IEEE Code of Ethics (IEEE COE), ACM Code of Ethics and Professional Conduct, and IEEE Computer Society (IEEE CS) / ACM Code of Ethics for Software Engineers,we have come up with my best mapping of ethics codes to each criterion. Table 7.2. contains the mappings along with my reasoning and how the code differs from the preexisting mapping to NSPE in McCormack et al. 's Table 1.

Area of Responsibility	**Code**	**Reason**	**Difference from NSPE**
Work Competence	ACM 2.1, 2.2, and 2.6	These codes stress the importance of performing work to the best of your ability and	ACM dictates the importance of high-quality work.

		being honest about your capabilities.	
Financial Responsibility	IEEE CS 3.09	This code states that one should ensure that the product is reasonably priced and have realistic cost estimates.	IEEE CS says to disclose a range of uncertainty when reporting cost estimates.
Communication Honesty	IEEE COE i.3-i.5	These codes state how one should disclose conflicts of interest, communicate criticism, and not accept bribery.	IEEE COE states that one needs to disclose conflicts of interest.
Health, Safety, Well-Being	IEEE COE i.1	This code makes it clear that safety, health, and privacy of the public is of utmost importance.	IEEE COE makes it clear that one has a responsibility to protect privacy as well.
Property Ownership	ACM 1.6 and 1.7	These codes talk about how one should uphold privacy and keep data collection to a minimum. One should also keep confidential information private unless there is a violation of the law or the code.	ACM offers more insight about data collection and how not to abuse trust.
Sustainability	IEEE COE i.1	This code references that one should comply with sustainable development practices and protect the environment. One should also reduce the amount of waste generated.	NSPE does not address sustainability.
Social Responsibility	ACM 3.1	This code says that how people will be impacted should be the main concern and should always be considered.	NSPE focuses on Honesty, where ACM focuses on benefit to society.

Table 7.2. The Seven Areas of Professional Responsibility Mapping to a Society's Ethics Code

Discussed below is our interpretation of each area.

Work Competence: The need to understand the work you are doing and the technology you are developing with how it will affect society and consumers.

Financial Responsibility: To be responsible financially, with research grants/companies money and yourself.

Communication Honesty: To be truthful to society and the consumers. Especially if a problem with the technology is discovered, whether it has physical problems and could be harmful or if there are data leaks.

Health, Safety, and Well-Being: It is important that the product is safe, that testing is done in a safe way and that consumers are safe when using the product and that they use it correctly. It is also important that the developers are safe and treated well.

Property Ownership: Owning the intellectual property of the product/design. Being responsible with its ownership, both with its development/continued development and allowing/disallowing licensing of it.

Sustainability: Making sure your product is sustainable and environmentally friendly as well. Making sure that it generates the minimal e-waste possible.

Social Responsibility: Responsibility that any materials needed are obtained ethically. Need for privacy and customer data protection. Should be used in a good/meaningful way.

7.2 Project Specific Professional Responsibility Areas

Now we discuss each areas' importance with respect to our project.

Work Competence (High): It is important to understand technical details of our project to get correct results. We would say we are mostly prepared at a technical level for this project, however, we still will need to learn how to work with the various frameworks, APIs, and visualizations.

Financial Responsibility (Low): Our project is completely software driven and has little to no budget. For our performance we don't have a grasp on the scale of the project even though we have not had any costs yet.

Communication Honesty (High): It is important to mention difficulties to our advisor and have good communication throughout our team. We have been working together for a couple of months now and have completed many documentation homeworks. From this we have gotten very comfortable with each other and we believe we can all communicate honestly with each other and will let everyone know if we disagree with any aspects of the project.

Health, Safety, and Well-Being (Medium): Although our project is completely software based and doesn't have direct health impacts, we still need to uphold our own personal health.

Property Ownership (Low): By default the project is owned by the university and our project will be utilizing open-source software.

Sustainability (Medium): We are creating a digital website. So we do not have to worry about any physical waste. The only thing required would possibly be a server, however we would not physically manage it so it is therefore out of our control.

Social Responsibility (High): We always need to be aware of the impact on people who use our software. We do not have a lot of experience with data privacy other than some basic encryption, so we would need more research if we want to store users' data.

7.3 Most Applicable Professional Responsibility Area

We agree that we have high Communication Honesty proficiency. This is important because this area greatly impacts the overall project and outcome. Open and honest communication within a group allows for problems to be dealt with quickly and in a professional manner. Our team has weekly meetings where we discuss project updates and any problems that need solving. We also go back to our client to show design documents and get feedback. This lets us be upfront and honest with where we are with the project and what we know. Because of this, our team has had good collaboration with each other, trust from our client, and problems are dealt with up front.

Even though we are excelling at Communication Honesty, our team needs to improve Work Competence. Our team has been lacking the experience in some relevant areas of our project such as Machine Learning. This can lead to an incorrect or poorly implemented solution to the client's needs. Going forward, our team will conduct more research into these areas and concepts.

8 Closing Material

8.1 Discussion

We are making a product to assist students and professors in developing their research ideas and gathering information and funding for research projects. So far, we have either met the design requirements or have plans to do so next semester.

8.2 Conclusion

Thus far, we have successfully completed the design for our project solution and implementation. We accomplished our goal of having our completed design document by the end of this semester. Going forward, our goals will be to ensure the functionality of our product at each implementation level. Iterating through these scenarios is our best plan of action. The plan is to start by implementing basic functionality, we will then add more complex functionality, and so on until we have fully implemented our design. This plan allows us to break apart the problem that is proposed by the project. Smaller problems will allow us to get work done more quickly and efficiently and allow us to test each part of our design.

8.3 References

X. Chen, F. Zhou, K. Zhang, G. Trajcevski, T. Zhong and F. Zhang, "Information Diffusion Prediction via Recurrent Cascades Convolution," 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2019, pp. 770-781, doi: 10.1109/ICDE.2019.00074.

F. Zhou, X. Xu, K. Zhang, G. Trajcevski and T. Zhong, "Variational Information Diffusion for Probabilistic Cascades Prediction," IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, 2020, pp. 1618-1627, doi: 10.1109/INFOCOM41043.2020.9155349.

D. Gotterbarn et al, “Code of ethics: IEEE Computer Society,” Code of Ethics | IEEE Computer Society. [Online]. Available: https://www.computer.org/education/code-of-ethics.

Mapbox. [Online]. Available: https://www.mapbox.com/.

8.4 Appendices

8.4.1 Team Contract

Team Members:

1) Bailey Gorlewski 2) Evan Gossling 3) Ian Johnson

4) Paul Brinkmann 5) Will Postler

Team Procedures:

1. Day, time, and location (face-to-face or virtual) for regular team meetings:

   Tuesday, 5:30 PM, hybrid

2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):

Combination of Discord and face-to-face for all updates, reminders, issues, and scheduling. Email if something is urgent or they are not responding to Discord and we do not see them face-to-face.

3. Decision-making policy (e.g., consensus, majority vote):

Group discussion with a following vote (majority rules) (wheel of spin if we can't decide)

4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):

Ian will update a shared file with meeting recaps, and make them available to other members in our shared folder.

Participation Expectations:

1. Expected individual attendance, punctuality, and participation at all team meetings:

Expected to attend and participate in all team meetings, unless given advance notice and given respective updates through Discord. Expected to be punctual, aka show up around meeting time and not 30 min late or later.

2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:

Team members will each contribute an "even" share of the work, and will complete work based on set timelines/deadlines. Team members will communicate if they have any stop gaps or blockers before the deadline. Team members will be expected to complete their assigned work before the deadline.

3. Expected level of communication with other team members:

   At minimum, 24 hour response to teammates via Discord.

4. Expected level of commitment to team decisions and tasks:

Contribute an equal amount of work as well as participate in each large-scale decision.

Leadership:

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):

Team Organization: Ian Johnson

Client Interaction: Will Postler

Component Design: Paul Brinkmann & Evan Gossling

Testing: Bailey Gorlewski

2. Strategies for supporting and guiding the work of all team members:

Communicating over Discord to ask questions where other team members can respond to questions and support other members

3. Strategies for recognizing the contributions of all team members:

Go over contributions in weekly meetings

Collaboration and Inclusion:

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.

Bailey Gorlewski:

- Worked with OOP programming in an enterprise software development life cycle, and learned strong documentation and testing methodologies. Strong communication skills and translation abilities for explaining processes to users who may not have technical knowledge.

Evan Gossling:

- Used OOP to write scripts for clients needs. Demoed scripts for clients and listened to their feedback to make necessary changes.
- Worked on several projects, one of which focused around web development.

Will Postler:

- I have 8 years professional experience working in a team setting, managing projects and working with clients.

Ian Johnson:

- Worked on several development teams using object oriented programming, and embedded systems.
- Quick learner and motivated to succeed.

Paul Brinkmann:

- Worked on several projects. Have leadership experience. Been programming since age 13. Bringing forward the bald perspective.

2. Strategies for encouraging and supporting contributions and ideas from all team members:

Brainstorming sessions will be had in meetings, in order to provide all team members an avenue to share their ideas and contributions.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

Communicate to the team during a team meeting if they have any blockers or feel like they aren't being heard. Each meeting will start with a brief discussion on team morale.

Goal-Setting, Planning, and Execution:

1. Team goals for this semester:

Achieve an A in course, have a successful project with a sound design. Increase our team collaboration skills.

2. Strategies for planning and assigning individual and team work:

Create a project plan/agenda and assign tasks based on personal interest, goals, and experience for each team member.

3. Strategies for keeping on task:

Having an agenda/GANTT chart that keeps a timeline and responsibilities

Consequences for Not Adhering to Team Contract:

1. How will you handle infractions of any of the obligations of this team contract?

Have a group intervention to see why they are not complying to the contract and what they need to do to start complying and what we can do to help them comply.

2. What will your team do if the infractions continue?

Bring it up to our advisor, and if needed our professor.

****

a) we participated in formulating the standards, roles, and procedures as stated in this contract.

b)we understand that we are obligated to abide by these terms and conditions.

c)we understand that if we do not abide by these terms and conditions,we will suffer the

consequences as stated in this contract.

1) Bailey Gorlewski DATE 9/13/2022

2) Ian Johnson DATE 9/13/2022

3) Evan Gossling DATE 9/13/2022

4) Paul Brinkmann DATE 9/13/2022

5) Will Postler DATE 9/13/2022