INTELLICHEF

Spring-2024

Team-3
Date of Submission-3/4/24

Team Members:
Vincent Gruse
Rahal Dathanarayana
Sowmya Sathi
Anthony Grill Jr.
John (Drew) Cook

CONTENTS

Brief Resumes:

Vincent Gruse -
Master of Science in Computer Science – Towson University (2025)
Computing Skills Inventory:

- Java, JavaFX, JavaScript, C++, Python, HTML, CSS, Git, Gradle
- iOS, Windows, Linux

Work Experience:

- Special Education Science Teacher – 10 years
- Personal Projects – 12 years
- Most of my computer science experience comes from developing embedded systems, mostly on ESP32, ESP8266, Arduino, and RaspberryPi based systems.

Sowmya Sathi -
Education:

- Master of Science in Computer Science – Towson University (2025)
- Bachelor of Science in Electronics and Communication Engineering – Jawaharlal Nehru Technological University (2023)

Technical Skills: C, Python, MATLAB, SQL
Present position: Research volunteer for a chatbot-based LLM under Professor Xin Huang, Towson University

Anthony Gillis Jr. -
Master of Science in Computer Science – Towson University (2025)
Bachelor of Science in Computer Science – Towson University (2021)
Computing Skills Inventory:

- Java, C++, HTML, JavaScript, SQL, CSS
- Windows, Linux

Work Experience:

- Front-End Developer – 6 months

John Cook -
Master of Science in Computer Science – Towson University (2025)
Computing skills inventory:

- Java, Python, C, HTML, CSS, R
- Windows, iOS, Linux

Experience:

- Supply Chain Manager – 2 years

Rahal Danthanarayana -
Computer Science Education:

- Masters in Computer Science (Towson 2025)
- Bachelor in information technology (Sri Lanka - 2017)

Computing Skills:

- Backend - Java, Spring-boot, Git, Maven

- Frontend- JavaScript, Angular, Python, HTML, CSS,
- iOS, Windows, Linux

Work Experience:

- Software Engineer – 6 years
- I am mostly experienced in developing full-stack enterprise applications. And I'm experienced in cloud computing and Mobile app development.

Scheduling and Planning Table for A1:

Assignee Name	Email	Task	Duration (hours)	Dependency	Due Date	Note
Vincent Gruse	Vgruse1@students.towson.edu	-Facilitator -Create GitHub repository as described -Work breakdown -Project topic -Brief resume -1/2 of Teamwork Basics (Ground rules & personal experience)	5 hours	Discord, GitHub	3/4/2024	Hard due date for Sowmya to complete formatting.
Rahal Danthanarayana	Rdantha1@students.towson.edu rahalncm@gmail.com	-Brief resume -System requirements	4 hours	None	3/42024	Upload this in the shared file.
Sowmya Sathi	Ssathi2@students.towson.edu	-Brief resume -Project topic -Report formatting	3 hours	None	3/4/2024	It must be completed in 1-2 days before the last submission for evaluation.
Anthony Gillis Jr.	Agilli12@students.towson.edu	-Brief resume -Problem statement	2 hours	None	3/4/2024	Upload this in the shared file.
John (Drew) Cook	Jcook29@students.towson.edu	-Brief resume -½ of Teamwork Basics	3 hours	None	3/4/2024	Upload this in the shared file.

Teamwork Basics Summary:

Within the domain of efficient cooperation, several fundamental concepts surface to support the completion of tasks and team member satisfaction. These core elements are covered in detail in this summary, which highlights the significance of knowing team members, establishing ground rules, fostering open communication, designating a facilitator, and skillfully managing typical difficulties.

Team Members:
Building a deep grasp of team members' abilities is essential to effective teamwork. Acknowledging personal potential helps create an atmosphere in which skills are best utilized.
Ground Rules:
Setting ground rules helps the team go in the direction of cohesion and productivity. These guidelines cover a wide range of topics, including how work is distributed, the roles of facilitators, communication techniques, meeting logistics, and meeting etiquette. While some norms could be clear from the start, others might develop naturally as the partnership progresses.

1. **Work Norms:**

   Responsibilities within the team will be assigned by the current facilitator. Members will have the chance to express preferences for specific project aspects. The facilitator will set deadlines and facilitate communication regarding any challenges with project requirements. If a member is unable to fulfill their obligations, timely communication is expected to enable collaborative problem-solving among the group. Disagreements regarding work quality will be resolved through a democratic process within the team, allowing progress on the project. Respect for individual work habits is paramount, provided they do not impede the progress of others. Members inclined to work better under deadline pressure are encouraged to select tasks that complement rather than overlap with those of their peers.

2. **Facilitator Norms:**

   a. The facilitator's role becomes increasingly important in leading the team toward productivity. In addition to being a leader, the facilitator also ensures that the group stays on course. Focusing on the job at hand, assuring participation, adhering to time limits, offering suggestions during delays, assisting in addressing issues, and assessing choices are just a few of the responsibilities. This function rotates to guarantee a variety of facilitation approaches.

   b. The project team will use a rotating facilitator for each project segment. Facilitators will be selected through voluntary participation and consensus among all team members. Their role will involve breaking down project sprints into manageable tasks and presenting them to the group. Additionally, facilitators will create a visual or collaborative document for tracking progress and documenting work contributions by group members. The facilitator will also ensure that all GitHub requirements are distributed and completed appropriately.

3. **Communication Norms:**

   a. Effective teams depend on honest and open communication to function. Establishing a trusting and cooperative environment starts with addressing objectives, priorities, and possible disputes head-on. Recognizing personal priorities, particularly in relation to grades and course priorities, can assist prevent conflict on the team.

   b. Communication will primarily occur through text messages on a Discord server. Group members should expect to check Discord message every 2-3 workdays to stay up to date with project progress. Calls will be scheduled on an as-needed basis, depending on the project's progress and specific questions or discussions among group members. Individual preference should be described within the server should a group member be unsatisfied with the current mode of communication.

4. **Meeting Norms:**

   The group facilitator will be responsible for scheduling meetings. This should include the time, place, and communication method (i.e. Discord call, Video call, Text, etc.). Group members unable to attend should check the Discord server for missed information and relevant project information.

5. **Consideration Norms:**

   The facilitator will be responsible for leading discussions during informal meetings aimed at advancing the project's progress. If any behavior hinders this objective, the facilitator should address it with the individual. Norms

are flexible and may evolve throughout the semester as the project develops. These adaptations should accommodate the unique needs of each group member, ensuring both the project's success and the satisfaction of all involved.

Hints for Handling Difficult Behavior:

- Managing difficult personalities in a group requires skill. Techniques include using comedy to deflect conversations that are overpowering, having private talks to address behaviors, bringing quieter people out, and responding positively to criticism.
- In my professional experience, I have encountered instances, especially in managerial roles, where team members exhibited difficult behavior. Addressing these issues in a constructive and non-confrontational manner has been key. For example, when a team member is not contributing equally, I have found it effective to have a conversation to express concerns and find ways to distribute tasks more effectively.
- Setting clear expectations by outlining roles, responsibilities, and deadlines from the onset of a project is crucial to avoiding misunderstandings and conflicts. When a team member consistently misses deadlines, a reminder of agreed-upon timelines and their significance can help reinforce accountability.
- In cases where difficult behavior persists and disrupts the project, involving a mediator, such as a supervisor or HR representative, can assist in facilitating a resolution. This external perspective can mediate conflicts and work towards a mutually beneficial solution for all parties involved.

Hints for Handling Group Problems:

- Group dynamics provide several difficulties, and effectively resolving them is essential:
    - Teams that are struggling gain from having well-defined goals and tasks.
    - Tangential conversations need to be brought back to the main task at hand.
    - Encouraging in-depth conversations helps reduce rash decisions.
    - Strategies like multi-voting and Plan A are used to address decision-making issues.
    - Conflicts must be resolved quickly and amicably through communication.
    - Promoting inclusivity and opposing ridicule helps address the problem of people making fun of or disregarding other people. To address the root causes of unequal work contributions, direct contact is necessary.
- In summary, a comprehensive strategy that incorporates comprehension, communication, and skillful problem-solving is necessary for effective teamwork. Task satisfaction and team member satisfaction are increased in a positive and productive teamwork environment when there is regular communication, flexibility, and cooperative problem-solving.
- In the workplace, it is inevitable to encounter challenging behavior from peers and managers due to differing perspectives. In my experience, the key to managing these behaviors effectively lies in maintaining open communication and ensuring transparency regarding expectations and outcomes. While this may sometimes feel awkward, it establishes a foundation of trust and fosters a sense of collective effectiveness within the team.

Problem Statement:

IntelliChef, at its core, is a recipe search engine with a focus on using machine learning to help provide a better experience in finding recipes based upon the user's leftover ingredients. IntelliChef will provide and maintain a user-friendly interface that will seamlessly allow users to navigate the site. Our product's customer base will be quite diverse ranging from casual cooks, people looking to try something new, professional chefs, or whoever wants to try out our product.

There are currently many alternatives on the market to IntelliChef. These include brands such as SuperCook, DishGen, and even Tesco's own Recipe Generator. A commonality with these brands is that they do not have a filtering system that allows for the removal of duplicate recipes. IntelliChef will stand out among its competitors based on the greater emphasis on machine learning techniques to quickly obtain and filter through recipes for users while removing any potential duplicates that may show up.

Our top-level objective for IntelliChef is to provide an easy and robust experience in finding recipes to cook. Convenience is, also, imperative to keep users engaged and to keep users coming back to use our service. From a technical point of view, IntelliChef has many interesting points such as our machine learning systems, our many filtering systems (Being able to filter by cuisine, recipe rating, and recipe time), and the multitude of connections between components within the product itself.

There are a multitude of resources and tools to help make IntelliChef a reality. We will be using Java Spring's framework and APIS to support our backend development, Angular framework to support our frontend devlopement, SQL to create and maintain our databases, and the development of our machine learning components will be done in Python. All these parts will be integrated seamlessly with each other to create a product that our end users will come to love.

System Requirements:

1. Introduction
This outlines the system requirements for the development and deployment of a full stack web application. The application will be built using Java for the backend, Angular for the frontend, and MySQL for the database management system.

2. Functional Requirements
2.1 User Authentication
- Users should be able to register for an account.
- Users should be able to log in using their credentials.
- Passwords must be securely stored and encrypted.

2.2 Data Management
- The application should allow users to create, read, update, and delete (CRUD) data.
- Data should be stored and retrieved from the MySQL database.
- Data integrity and consistency must always be maintained.

2.3 User Interface

- The frontend should be developed using Angular framework.
- Front-end should use modeler approach since it will make the application lightweight.
- The user interface should be intuitive, responsive, and user-friendly.
- Proper error handling and validation messages should be displayed to users.

2.4 Security

- The application must implement proper security measures to prevent unauthorized access and data breaches.
- Input data should be validated on both the client and server sides.
- Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF) attacks should be mitigated.

2.5 Performance

- The application should be optimized for performance to ensure fast response times.
- Database queries should be optimized to minimize latency.
- Caching mechanisms should be implemented where necessary to improve performance.

3. Non-Functional Requirements

3.1 Scalability

- The application should be designed to handle many concurrent users.
- Horizontal and vertical scalability should be considered during the design phase.
- Using Docker for containerization and consider docker-swarm or Kubernetes for container orchestrations.

3.2 Reliability

- The back-end should implement proper error handling (checked and unchecked exceptions) and should communicate the error clearly.
- The system should be reliable with minimal downtime.
- Backup and disaster recovery mechanisms should be in place to prevent data loss.

3.3 Compatibility

- The application should be compatible with modern web browsers such as Chrome, Firefox, and Safari.
- The backend should be compatible with Java EE specifications.
- The database should be compatible with MySQL version X or later.

3.4 Development Environment

- Developers should use Integrated Development Environments (IDEs) such as IntelliJ IDEA or Eclipse for Java development.
- Angular CLI should be used for Angular development.
- MySQL Workbench or similar tools can be used for database management.

3.5 Deployment Environment

- The application should be deployable on popular cloud platforms such as AWS, Azure, or Google Cloud Platform.
- Continuous Integration and Continuous Deployment (CI/CD) pipelines should be set up for automated deployment.

4. Constraints

- The development team should adhere to best practices and coding standards for Java, Angular, and MySQL.
- Any third-party libraries or frameworks used in the application should be properly documented and licensed.

5. Glossary

- CRUD: Create, Read, Update, Delete
- XSS: Cross-Site Scripting
- CSRF: Cross-Site Request Forgery
- IDE: Integrated Development Environment
- CI/CD: Continuous Integration and Continuous Deployment

Appendix:

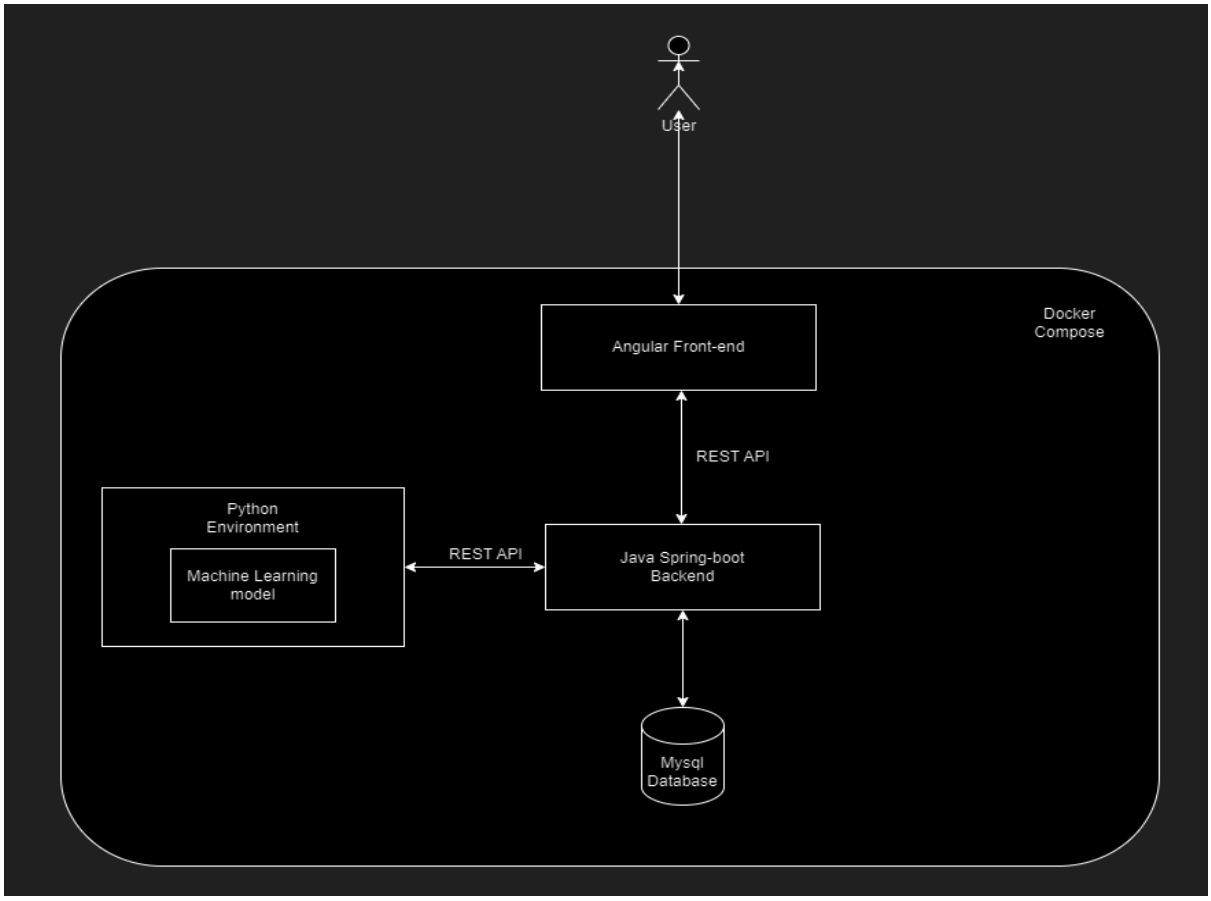

Figure 1: System Architecture

Link 1: https://github.com/vincentgruse/COSC612-IntelliChef

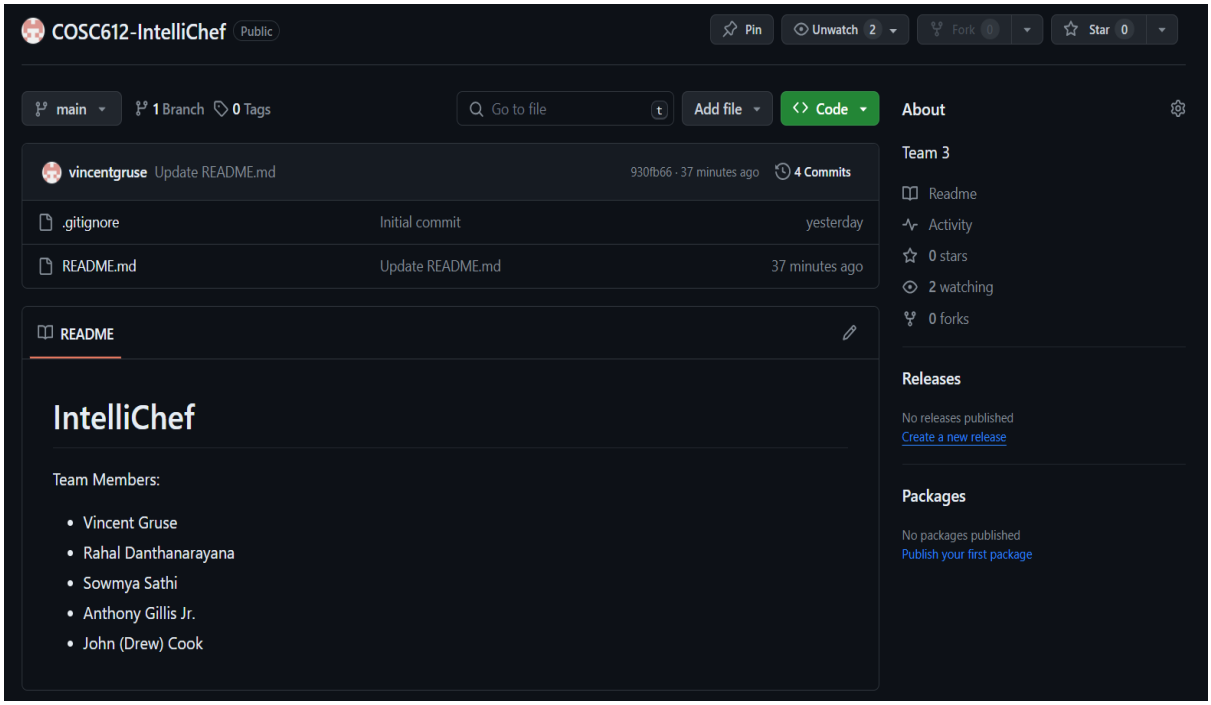

Figure 2: IntelliChef Github README

Link 2: https://github.com/users/vincentgruse/projects/1/views/1

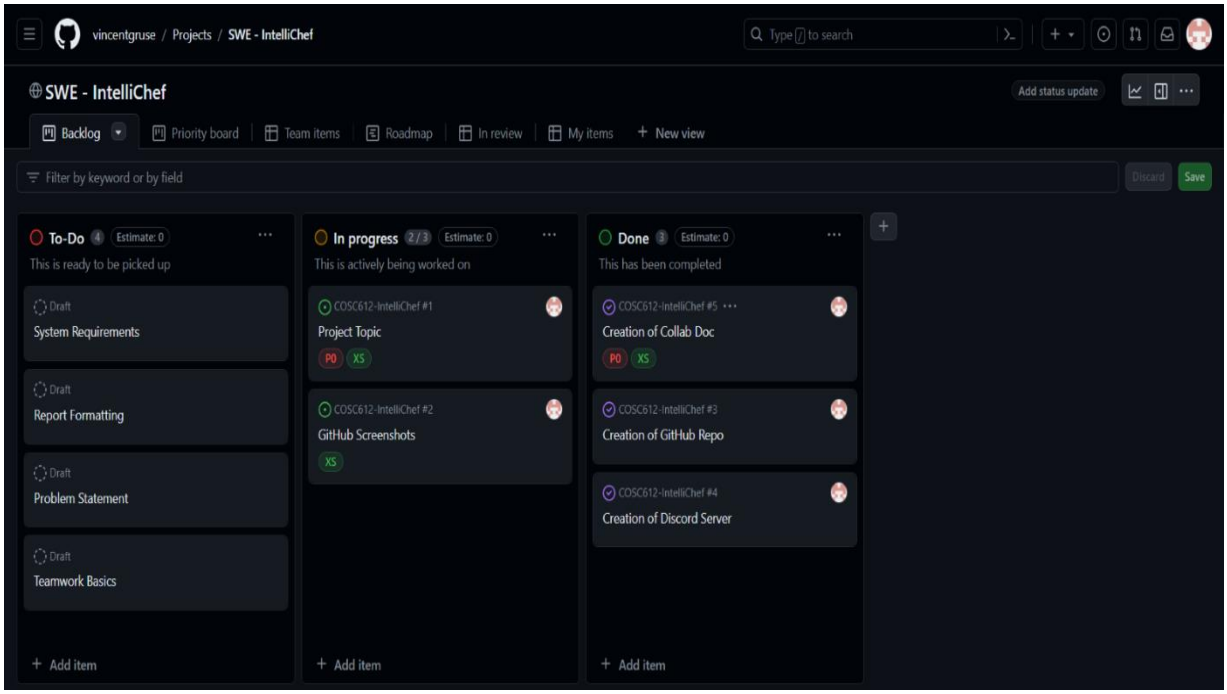

Figure 3: IntelliChef Project Board