Team Name:
🥐Scrum-ptious🥐

Group Picture:
[redacted]

Team Strengths:
Java, C/C#/C++, Kotlin, Groovy, Python
PHP, React, Javascript, Tailwind, Bootstrap, HTML
MySQL/MariaDB, MS SQL, SQLite, PostgreSQL, MongoDB, Redis
Bash, Git, GitHub Actions, Maven, Gradle, Ant

Individual Team Member Introductions:

[redacted]: I am a Computer Science Honors student with a minor in Mathematics and a concentration in Cybersecurity. Through my coursework, I have built a strong foundation in programming languages such as Java, C, and Python. More recently, I have expanded my skill set by exploring web development using tools like HTML, CSS, and JavaScript. Beyond academics, I work as a manager at Kumon, where I have developed essential soft skills, including leadership, communication, adaptability, and conflict resolution. This experience has strengthened my ability to collaborate effectively, solve problems under pressure, and mentor others—a skill set that complements my technical skills.
GitHub: [redacted]
LinkedIn: [redacted]

[redacted]: I am a machinist-turned-student seeking to transition into a software engineering position. I have been programming as a hobby for 14 years, wearing a variety of hats as necessary and doing a little of everything from manually writing G-code to systems administration. A lot of my hobby work involves reverse engineering lightly-obfuscated code, and the most fun I had in the past year was a deep dive into implementations of bounding volume hierarchies. The biggest feather in my cap is OpenInv, a project I inherited in 2017 after actively maintaining it since 2014. It's currently at 3,800,000 downloads and counting!
GitHub: [redacted]
LinkedIn: [redacted]

[redacted]: I am a Computer Science Honors student with a strong passion for mathematics, problem-solving and software development. Proficient in Java, C, JavaScript, and Visual Basic, I specialize in applying object-oriented principles in both front-end and back-end development. My academic foundation in Data Structures, Algorithms, and Machine Learning, combined with hands-on projects using Weka, has prepared me to optimize time and space complexity and perform data analysis. A quick learner and responsible team member with strong project management skills, I am eager to contribute my technical expertise to collaborative software engineering projects.

GitHub: [redacted]
Linkdeln: [redacted]

[redacted]: As a computer science student with a mathematics minor, I am passionate about technology and problem-solving. I have hands-on experience in web development, including PHP, JavaScript, and some React, along with a solid background in database design using SQL. Proficient in Java and Python, I also have a keen interest in algorithmic analysis and have conducted research in machine learning. Beyond technical skills, I'm a strong communicator who thrives in team settings. My management experience from my part-time job has sharpened my ability to delegate tasks and lead effectively. I'm excited to bring my skills to innovative and impactful projects.
GitHub: [redacted]
LinkedIn: [redacted]

[redacted]: **PRIMARY COMMUNICATOR** I am a Software Engineering student at Central Connecticut State University with a passion for automotive manufacturing and software. My interest in software began with Boston Dynamics and grew into a career path after witnessing SpaceX's first successful self-landing Falcon 9 rocket. Through my studies, I have developed strong programming skills in Java, Python, C, HTML, CSS, Blazor Web Framework, and SQL. My focus areas include object-oriented programming, web development, and system architecture. I have worked on projects involving full-stack web applications, responsive UI design, and front-end/back-end integration to develop efficient, scalable solutions.
GitHub: [redacted]
LinkedIn: [redacted]

Virtual Meeting:
[redacted]

Team Agreement:

Methods of Communication	- Internal group communication: Discord (primary for group chat discussion and quick updates) - Formal client communications: Email (ensure that all members are included in CC)
Communication Response Times	- Within 3 hours during working hours (9am-4pm, weekdays)

Meetings	- Internal group communication: for in-person group meeting weekly
	  - Location: MS 208 (or another quiet room)
	  - Schedule: 3:15pm - 4:25 pm every Monday
	  - Attendance: not mandatory for all members since those unable to attend should stay updated via Discord.
	- Formal/client communication:
	  - Virtual meetings will be held via Microsoft Teams during available slots on M/W 3:00pm-4:30pm.
	  - In-person meetings: to be scheduled as needed.
	- Minutes/notes should ideally be collaborative - if you have an idea, write it down. Notes will be saved in Google docs.

Meeting Preparation	- Team members should review the previous meeting notes and put forth their best effort to complete the assigned tasks before the meeting. - As meeting content will be loosely determined in advance, members should be prepared to speak to any problems, solutions, or ideas in areas they are responsible for.

Version Control	- Use .gitignore and other configurations to prevent unnecessary files in the repository. - Code style: Adhere to client requirements, establish own conventions if necessary. Publish if possible, i.e. .gitattributes and .editorconfig files. - Commit messages need to be descriptive. - Prefer machine-readable commit convention (i.e. Conventional Commits), but not a hard requirement. - Don't commit nonfunctional code to the main branch unless necessary. Prefer merging into other branches as needed until CI and unit tests pass. - Prefer preserving full history over a squash when merging.
Division of Work	- Use scrum to estimate time required to perform tasks and attempt to divide tasks evenly. - Try to ensure that members get to work with every aspect of the project to ensure that all members are cross-trained on any technologies and have a greater understanding of the project as a whole.
Plans for collaboration among the team members	- Team will collaborate with the utmost respect and assist each other both in person and online - Monday/Wednesday: Longer informal meeting/collaboration 3:15pm-4:15pm - Evening online check ups through discord will be arranged when necessary
Submitting Assignments	- Submission roles: Primary: Erik; Backup: Adam - Review process: Before submission, all team members must review and approve the submission. - Submission deadlines: At the very least, whatever the agreed/assigned due date. However finishing the assignment 12-24 hours is ideal to allow time for review and adjustments before submission of deliverables.

Contingency Planning	- Have a calm and civil discussion about what has happened to understand the issues but also hold the person accountable. - Focus on solutions rather than negativity. Keep communication open and honest. - Communicate with the instructor for further guidance - In the worst case, if someone drops out, we will adapt to ensure our project stays on track. We were originally prepared to be a group of 4 based on the initial requirements outlined.

Write-up and reasoning:

The team will use email for client interaction to document and record communications. The team already utilizes Discord for academic and informal communication. A 3-hour response window allows for the fact that the team is attending classes while still being relatively responsive. While the team is likely to meet more regularly due to common classes, having scheduled meetings will keep members on track for individual deliverables. Attendance is heavily encouraged but not mandatory; updates will be shared via Discord and minutes will be available for review. Minutes will be collaborative and stored in Google Drive to ensure that all relevant details are recorded. Client meetings will occur during normal class time. The team should be prepared with any questions or concerns in advance as well as being able to speak to areas of responsibility. The team discussed various approaches to version control and agreed on a standard to follow, ensuring that the repository will be uncluttered and the commit history will be complete. As the team expects to be working with an existing code base, the team plans to match the existing formatting and code style. The scrum methodology will be used to estimate and distribute tasks evenly. Members will rotate working on various aspects of the project to ensure cross-training and prevent a degradation of team capabilities in the worst-case scenario where a team member is lost. Collaboration will be conducted respectfully, and any issues will be addressed through discussions while ensuring accountability. Should issues arise, the team is committed to communicating openly and calmly to resolve any problems in a constructive fashion. Submissions will be handled by [redacted], or [redacted] if [redacted] is unavailable, with all members reviewing deliverables 12-24 hours before the deadline.