Caitlin Graves, Molly Iverson, Skyllar Estill, Sierra Pine, Naomi Dion-Gokan 02/02/2023

# Team Name: MCSSN – LibraryAssistant

For this milestone, we are providing information on the key tools, technologies, and process model used for our Library Assistant course project.

## Key Tools:

- Microsoft Visual Studio Code (an IDE to write the code)
  - Reasoning: Everyone in our group has experience coding with Visual Studio. Visual Studio has many built-in tools and abilities that will help us throughout the process of developing software. We will be using C++ to code our project.

- Crow
  - Reasoning: We need web application framework software compatible with C++ to build our web application. This software provides libraries and resources. Crow has more online tutorials than similar software. C++ is not typically used for web development, so it is practical to use a framework.

- GitHub
  - Reasoning: Along with Visual Studio, we all have experience with GitHub. GitHub keeps track of our commits and what the group adds. Since GitHub is where we will post our project, TAs and the professor can see what we are doing.

- Figma
  - Reasoning: Figma is a good web design interface to visualize the website/app. Our group will see the look and feel of the WebApp before we build it. The aesthetics of a WebApp can be just as important as the technical design. This interface will help us visualize the same idea before constructing it.

- Microsoft Word
    - Reasoning: We will use Word to document our progress and requirements to fulfill. We will keep track of individual responsibilities and questions to discuss at our next meeting. Microsoft Word contains features to make our documentation visually appealing and easy to understand.

- Netlify
    - Reasoning: We are using this cloud platform to host the website so that other people can access our final project (not code).

- Discord
    - Reasoning: We are using Discord as a main source of communication to easily send messages from our computers and phones. It is also useful for sharing documents, links, and videos.

- Draw.io (or some other UML tool)
    - Draw.io is a free, online diagramming and flowchart software. It offers an intuitive interface, a wide range of templates, and a vast library of shapes to make it easy to create professional-looking diagrams and charts. Draw.io supports real-time collaboration so our team can easily use it with platforms like Google Drive. We can use the program on both Mac and Windows, which is useful since we have different operating systems in the group.

**Technologies:**

- C++
  - Reasoning: C++ is the primary language we will use to implement each piece of our library assistant. All group members are on the C/C++ track, so C++ is the language we are most familiar with and comfortable using.

- HTML/CSS
  - Reasoning: HTML/CSS are essential to set up our web application. We researched the resources on building web applications in C++, and all of them included the use of HTML/CSS. CSS will be more helpful with the webpage design, while HTML will provide the structure.

- GitHub
  - Reasoning: Using GitHub, everyone on the team can easily access each build of our project. We can also keep track of commits and have version control. The TAs and the professor can monitor the state of development.

- Google Books API (application programming interface)
  - Reasoning: APIs are a way to communicate information between software. The Google Books API provides a straightforward way for searching up books with data such as the title, author, and even a picture of the book cover provided. Using this API, we can query books online without downloading a large database.

Caitlin Graves, Molly Iverson, Skyllar Estill, Sierra Pine, Naomi Dion-Gokan 02/02/2023

## Process Model:

**We will be using the spiral development model.**

This model couples the iterative nature of prototyping with the controlled and systematic aspects of the waterfall model.

Reasonings:

- We chose the **spiral model** as it is ideal for risk handling. Given that this project is a significant leap from previous assignments, there are a lot of unknowns and room for mistakes. There is plenty of risk for error during the development process. In every phase of the spiral model, we will analyze the product features and identify risks. The spiral model will help us deal with new bugs more effectively.

- This model does not require us to produce an executable for every cycle. We can devote a few cycles to requirements, design, and other activities. These gradual releases align with the class milestones and allow us to focus on refinement.

- We are aware of the drawbacks of this complex model. Defining objective milestones and estimating the number of required cycles can be difficult. The class milestones will create structure in our project. We will combat this by meeting consistently and spending considerable time planning. We will meet with Professor Cai during office hours to check our progress when needed.

- We did **not** choose the **prototyping model** as we understand customer requirements well. Two of our group members work at the WSU Holland and Terrel Libraries. We are also familiar with similar library assistance tools such as WSU Search It and Ex Libris Alma library management software.