Double Espresso Team Charter

Martin Munizza
Vanessa Rupertus
Stephanie Backovic
Jared Allen
Daniel Goettel

Document History		
Release No.	**Date**	**Revision Description**
1	7 February 2022	Initial release (Deliverable 1)

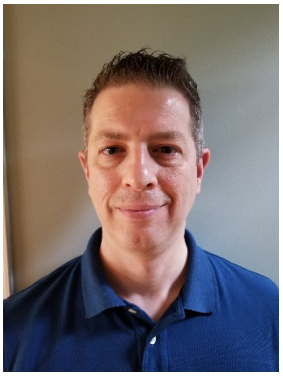
1. Team Name and Logo

Team Name – Double Espresso

Team Logo – Our team logo expresses our commitment to both speed and quality and highlights our strong ties to application development in the Java language:

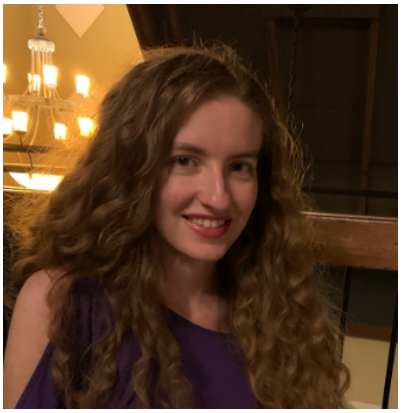
DoubleEspresso

2. Member Introductions



I'm Marty Munizza, team member for the Clue-Less project at Double Espresso. I possess a BSEE from Virginia Tech and I currently work for Lockheed as a Network/Security Engineer on US Navy and Coast Guard Information Systems Contracts. I'm currently enrolled in the MS Information Systems Engineering program at Johns Hopkins and have written Java code for several projects related to my coursework.
My professional work over the last five years included the following assignments:

- Design and evaluation of redundant and secure networks for tactical system baselines
- Verifying Network/Cyber security compliance with applicable government guidelines including leading remediation/redesign efforts to improve security posture of each baseline.
- Define/refine Local Area and Satellite network subsystem requirements
- Managed onsite teams of OE/Network/Test engineers' efforts to update shipboard electronic systems in preparation for builder's trials.
- Served as Integrated Product Team (IPT) Lead of a team of Operating Environment and Network Engineers to ensure new features, hardware, and capabilities were successfully delivered into system baselines.



I am Vanessa Leigh Rupertus. I graduated from Washington College (WAC) in '21 summa cum laude with a B.A. in English and minors in Creative Writing; Justice, Law, & Society; and Computer Science. Most of my academic career has consisted of analyzing novels/short stories and tutoring fellow students on constructing well-crafted and thought-inducing papers ranging from all subjects. After graduation, I began to take classes at JHU in the MS Computer Science program. During my academic career at WAC, I discovered a passion for Computer Science. I learned Object Oriented Programing, Database Systems, basics of Software Engineering and AGILE (Scrum), and more.
I have worked in C++, Java, SQL, MySQL, NoSQL, and have some experience with Python, JavaScript, and assembly language. I was also a finalist in the National Security Scholars Program (NSSP), where my employment fell through due to the initial quarantine.

Some notable projects I have worked on since my shift to Computer Science:

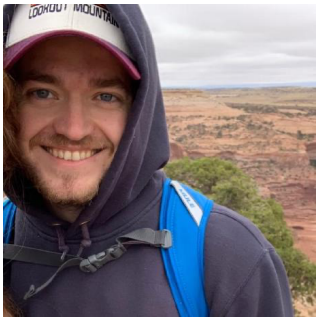
- A program that queried a database and returned for sale houses based upon user specifications
- An application (SafeEater) that allows users to tailor their profile to search for restaurants with specific accommodations, including specifications of COVID-19 safety protocols
- An independent study of MongoDB and NoSQL, where I created a product database

Stephanie Backovic here and I am a member of the Double Espresso Team. Currently, I am a patent attorney with a focus on drafting patent applications and opinions within the life sciences sector. I obtained my Ph.D. in Neuroscience from Weill Cornell and my J.D. from Cardozo. I realized my passion for CS when I noticed a recent paradigm shift in the field of biomedical research, as AI-based algorithms harnessed data in a much more powerful way. In the fall of ’21, I matriculated to JHU’s MS in Computer Science.

Technical experience relevant for the Clue-less project: I have worked with several programming languages, including Java, C++ and Python. I have a working knowledge of the software-hardware interface, which I gained through JHU coursework.

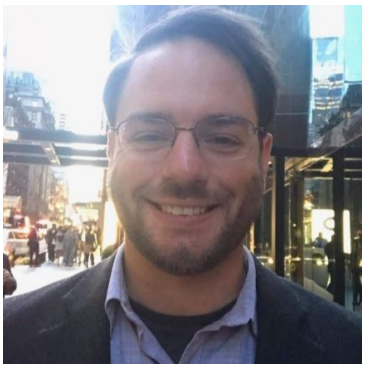
Exemplary projects through JHU coursework include:

- creating a Blackjack Simulator
- converting prefix to postfix expressions by generating stacks (using iteration or recursion)
- writing and comparing the performance of various iterative and recursive sorting algorithms



My name is Jared Allen, repping Double Espresso. I work as a Java developer for the Johns Hopkins University Applied Physics Laboratory. My bachelor's degrees are in Mathematics and Computer Science. I like that software engineering enables me to build real products for real people and provides a productive application for years of mathematical training. I want to design and implement complex applications that handle complex problems. I also enjoy spending time with my wife (we make a mean supreme pizza after countless Friday pizza-and-movie nights), reading (lately I’ve been interested in Russian literature), and building personal coding projects. I thoroughly enjoy designing software using object-oriented paradigms and design patterns. My personal projects include:

- A Tetris implementation in C++ using SDL2
- An expansion of the engine used to build Tetris, with rudimentary physics
- A small text editor, reusing much of the same code as the previous two projects



I am Daniel Goettel, the fifth Beatle of the Double Espresso's. I have an MS in Environmental Engineering from the University of Maine. One of my professors at UMaine was adamant that all engineers needed to learn a programming language, which was some of the best advice I received.

Currently I work as an environmental engineer in the field of pollution remediation, cleaning up polluted sites. During my engineering work I have been lucky to be involved in some large-scale statistical and data processing. I have also begun to move more seriously into data processing automation. These have included:

- Mann-Kendall Statistical Analysis with high throughput capability (MATLAB).
- Automated Hazardous Waste Tracking System (Excel VBA)
- Groundwater Pressure Transducer Data Processing System (Python)

So far at JHU I have completed Introduction to Programming Using Java, Computer Organization, and Data Structures. I am comfortable programming in Java and Python. I have taken two courses in Python from MIT EDx called Introduction to Computational Thinking and Data Science, and which I recommend as low-cost ways to learn Python.

3. Member Roles and Responsibilities

Project Manager – Vanessa Rupertus

The Project Manager guides discussions during group meetings and proposes deadlines on tasks, which the rest of the team will vote upon to set the deadline in stone. Their role is also to ensure the team discusses each deliverable in the timeframe specified in the Team Communication Processes section and to ensure each method of action is organized and planned accordingly. The Project Manager is expected to keep the team on task and ensure each deliverable is completed and submitted by the submission deadline. Finally, the Project Manager is responsible for maintaining a Work Breakdown Structure tailored specifically to the Clue-less project to ensure each deliverable is broken down into components that the team can tackle based on priority level.

Other Tasks: Outside of the Project Manager role, Vanessa is expected to communicate with other roles to determine where extra hands are needed. Extra tasks may include coding aspects of the project and ensuring quality of the software produced, for example.

Lead Architect – Daniel Goettel

The software architect is responsible for leading high-level design of the program. High-level design includes the overall use of modules and objects, and their proposed interaction. Specific deliverables may include Process Flow Diagrams and other architecture diagrams. The main function of the program architecture is to provide the Lead Programmer with a starting point and a plan, and to provide better organization for the coding process. As such, it is critical that the Lead Architect include the entire team in the process and obtain buy-in in any decisions and in the final design. Obtaining consensus from the coding team (which includes all team members) will lead to best results in implementation.

Other Tasks: Outside of the Lead Architect Role, the Lead Architect is expected to provide coding and testing support, and to use the tools specified by the Lead CM Engineer. The Lead Architect will cooperate with the project manager in supporting the team.

Lead Programmer – Jared Allen

The lead programmer is a resource for any programming issues or tasks the team has and aims to foster an environment where all team members can contribute to the project implementation. This includes advising on software best practices, guiding the implementation details of the architecture, and reviewing code. It also includes guiding the management of any code in the project, which may include automated tests, infrastructure, and version control.

Lead SQA (Software Quality Assurance) Engineer – Stephanie Backovic

The SQA is responsible for examining software function, design, content, and user friendliness throughout the product development process. This includes identifying potential issues by analyzing product requirements, as well as managing or participating in software testing. If any defects are identified, SQA is responsible for managing the change process by documenting defects and sending the software back to the developers for change.

Other Tasks: In addition to serving as SQA, Stephanie will communicate with other team members to ascertain if help is needed in other areas. Further tasks may include coding or testing, as well as helping with aspects of project management to facilitate product release.

Lead CM (Configuration Management) Engineer - Marty Munizza

The Configuration Management Lead will be responsible for selecting, implementing, and monitoring processes that control and document changes to the software product throughout its lifecycle. Changes usually result from a maturing design, correction of defects, or requirements undergoing refinement. The CM Lead is also ultimately responsible for verifying the completion and correctness of artifacts such as documents, drawings, and integrity of any submitted source code.

Other Tasks: Along with serving as CM lead, Marty will contribute to requirements development and analysis, with a focus on delivering sound, testable requirements as part of an overall test plan artifact.

In addition, Marty will contribute code modules, participate in testing software, and plan product demonstrations for key project stakeholders.

Lead Tester – Testing will be a shared responsibility among teammates

The Lead Tester is responsible for creating and executing a plan to verify and validate the requirements for the software product. Verification involves testing, demonstrating, or inspecting the release candidates for compliance with requirements; or, in other words, determine if the product was built correctly. Validation involves ensuring that the delivered product meets stakeholder expectations for product capabilities and functional characteristics, in other words was the correct product designed and built. The Lead Tester is also tasked with determining if requirements are functional or non-functional, and ensuring that accepted requirements are feasible, unambiguous, verifiable, consistent, and traceable.

4. Team Communication Processes

Double Espresso communicates primarily through Microsoft Teams, with email and Zoom as secondary strategies should technical issues ensue with Microsoft Teams. All teammates are expected to supply updates on their progress in their specified tasks of the project, and the team convenes through video calls whenever deemed necessary. Double Espresso utilizes Office 365's Word application to keep any document drafts for easy access and live updates.

By day two of each module, the team will discuss the next deliverable and decide upon a plan of action through video chat. Teammates are encouraged to seek early feedback about their ideas, to allow incorporating suggestions from other group members and to avoid re-work. In general, with each deliverable, there will be an initial meeting to discuss the task broadly and strategize. Following the meeting, members will perform the tasks assigned to them, and present the results to the group for final approval. Final approval will take place approximately 5 days before the due date (or 2 days in the event of a one-week deadline), to allow for last-minute corrections.

In the event of a disagreement about aspects of program design should arise, the problem should be brought up for discussion with all members of the team. Hopefully, a consensus can be reached. If not, the matter will be decided by majority vote. Additional details on how Double Espresso handles disagreements are described in the Conflict Resolution Processes section.

5. Conflict Resolution Processes

In the unlikely event a teammate is not performing their role as desired and other team members take notice, the rest of the team will convene and discuss the neglected duties of the teammate. After the team has identified the aspects of the project that are hindered by the teammate's performance, the entirety of the team will address the teammate in a scheduled meeting. The team will be respectful and firm in their approach and outline the points to which the teammate has not delivered. At the conclusion of the meeting, the teammate will then be given a specified amount of time decided upon by

the team to improve their performance. After the specified amount of time, the team will discuss the teammates' efforts of improvement. If an improvement is not made, the team will reconvene and determine if a dismissal is in order. A unanimous vote is required for a teammate to be dismissed.

Fostering a friendly and encouraging work environment is of the utmost importance. Should a teammate's behavior endanger Double Espresso's work environment, the team will convene in a similar manner to the potential dismissal process. The team will explain to the teammate how their behavior is detrimental to an open and friendly environment and the teammate will be asked to improve their behavior. Should that not occur, the team will decide what next steps to take, potentially including dismissal.

For conflicts that are not related to team member performance, such as competing ideas for project direction, or resolution of defects, or how to implement product features, Double Espresso intends to implement a collaborate or compromise model. We value a spirit of collaboration in solving problems but realize that not all situations will allow projects to move forward without strong differences of opinion. In such cases, the PM will mediate a fair compromise among the competing team members, such that all parties can be satisfied even if they are not completely happy with the outcome. The compromise would then be ratified by majority vote.