Problem Statement/Background

Problem Statement

Blackboard is a **Learning Management System (LMS)** used by universities nationwide as an online learning environment. **The University of South Carolina (USC)** uses Blackboard for most of its courses, making the software and knowledge around using it essential. This is why the university's **Division of Information Technology (DoIT)** is looking to create an engaging set of instructions to teach users how to submit assignments on Blackboard. After the user goes through these instructions, they will be prompted to take a short "quiz" to test the student's knowledge.

Who We Are

CyberEducation is a group dedicated to bringing learning to the modern age. Established in September 2021, we are a small group of 4 developers and designers that focus on delivering technology to empower both students and educators. CyberEducation was created by USC alumni who want to make learning and technology accessible to everyone. We pride ourselves on creating a user experience that is accessible and engaging. We have experience with creating educational software and developing learning management systems.

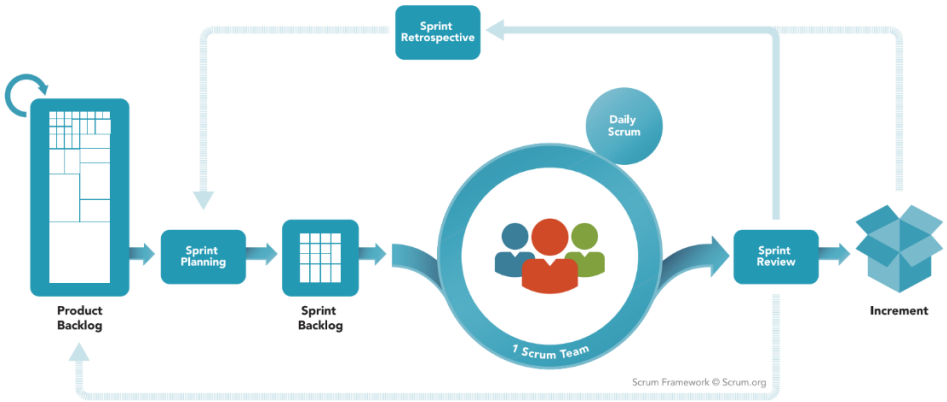
Methods/Procedure

Method Overview

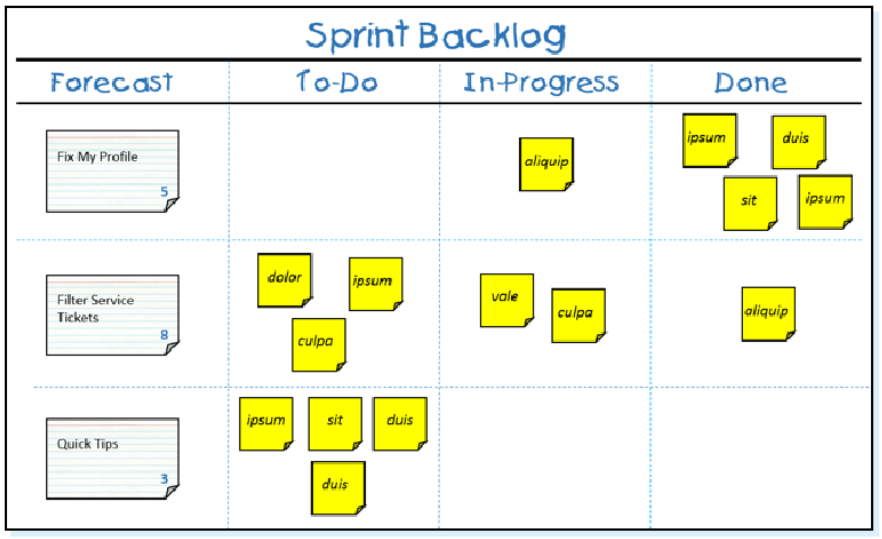
Our plan for this project is to create a program to act as a "game" version of Blackboard. The software will guide users through submitting an assignment through Blackboard. The software will be accessed through a Blackboard course, so the software will not need an additional server to host it. The software will give the user points for completing tasks. The software will link with the Blackboard course and allow the user to submit assignments through it.

What is SCRUM?

To make sure that our program is the best it can be within the given period, we will be using the SCRUM model. **SCRUM** is a simple, lightweight framework to insure effective teamwork and collaboration for complex projects.



Each week is divided into **"sprints"** with a set goal in mind for each week. The work is divided among the team in backlogs. The **product backlog** is where all tasks for the project are listed. The team decides what tasks are done each week and moves those tasks into the **sprint backlog.**



Source: scrum.org

The team has one large meeting at the beginning of the project known as a **sprint planning session.** These are long methodical meetings that break down every single detail of the project and what needs to be done. This allows the team to focus on product development later in the project. During development, the team will meet for 15 minutes a day for a **SCRUM meeting**. During the SCRUM meeting, the developers say what they have worked on the previous day, what they plan to work on that day, and what problems they are having in development. This allows the developers to collaborate with each other in an orderly way, without having to worry about abrupt constant meetings cutting into development time.

After the sprint is done, the team and the clients meet to discuss the work that has been done so far in the **sprint review** and discuss potential next steps so that the team is ready for the next sprint planning meeting in the **sprint retrospective.** For a short-term project like this, we plan to combine the sprint review and sprint retrospective into one meeting at the end of each sprint.

Procedure Outline

The following is a general outline of the procedure we intend to follow. It is divided into four sections that repeat each sprint.

Planning

The team will schedule meetings with the client about what specification the program requires. These requirements will then be written down in order of importance, then added to the product backlog. The sprint planning, sprint review, and sprint retrospective meetings happen during this time.

Coding

This is where most of the development will take place. Our team will be using the programming language **C++** to develop this program. Our code will be shared through a private repository on **GitHub**, a file hosting service for software developers.

Debugging

When the product is in a testable state, the team will start working out any issues or **"bugs"** that the project might have. Through a process called **debugging**, we go through our code and see where the program can potentially not work as intended or stop working entirely. We do this by creating extensively detailed tests called unit tests. We create a large variety of tests to ensure that every user interaction is bug-free.

Quality Assurance Testing (QA Testing)

When the team has thoroughly looked over the product, it will then be sent over to a group of students for outside testing. This is to test the usability of the software from a human perspective. Our testing group will be a volunteer group of 95 first-year students from USC. The students will be paid $25 dollars as compensation for their time and feedback. When we receive feedback from the QA testers, we will then take it back with us in the next Sprint planning to discuss what changes need to be made.

Qualifications and Resources

The Team

Joseph Oquinn - Software Developer

Education:

- B.S. Computer Science from the University of South Carolina
- M.S. Computer Science from the Kettering University

William Browning – Software Developer

Education:

- B.S. Computer Science from the University of South Carolina

Riley Paris – Lead Designer

Education:

- M.A. Service Design from the Savannah College of Art and Design
- B.S. Media Arts from the University of South Carolina
    - Minor in Computer Science

Nick Vector – Project Manager

Education:

- B.S. Computer Information Systems from the University of South Carolina
    - Minor in Business Information Management
- Certified Professional Scrum Master II

Previous works

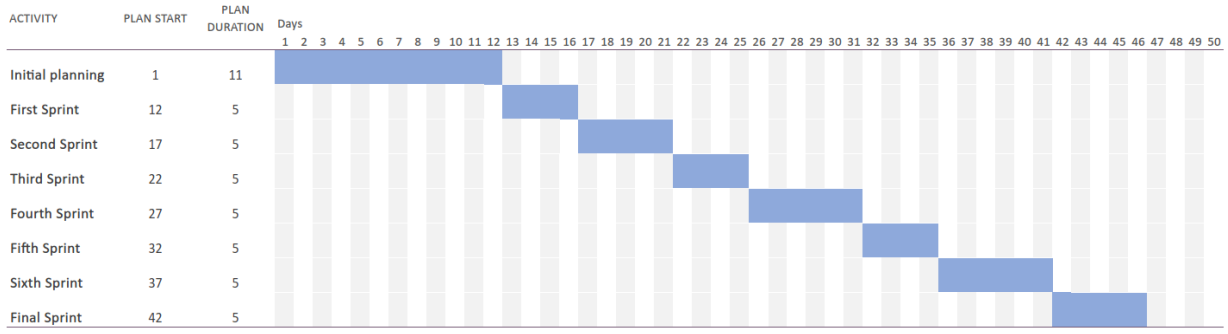
Our team has experience developing software to modernize education through technology.

- CookieCoding: A Learning Management System to help teach students how to code in various programming languages. Courses are created in the LMS by verified professors.
- CyberLibrary: A digital library platform that provides access to a wide range of educational resources, including e-books, videos, and interactive simulations.
- Linguist.ai: An artificial intelligence software designed to help the user practice their language skills.

Work Schedule

	Start date	End Date
Initial planning	05/01/2023	05/12/2023
First Sprint	05/15/2023	05/19/2023
Second Sprint	05/22/2023	05/26/2023
Third Sprint	05/29/2023	06/02/2023
Fourth Sprint	06/05/2023	06/09/2023
Fifth Sprint	06/12/2023	06/16/2023
Sixth Sprint	06/19/2023	06/23/2023
Final Sprint Before Release	06/26/2023	06/30/2023

Proposed Project Timeline



Grant Project Planner Template from Microsoft Create

Budget

Expense Description	Cost Per Person (USD)	# of People Paid	Cost total (USD)
Developer Pay	$1,500	4	$6,000
Equipment Cost	$906.25	4	$3,625
Volunteer Pay	$25	95	$2,375

Total: $10,000