WARE

Web Application for Robotics Education

University of Nevada, Reno

Department of Computer Science and Engineering

Revised Concept & Project Management

Team 17

Zachery Wiles, Ryan Lunt, Sean Griffith, Herman Hira

Instructors: David Feil-Seifer and Devrin Lee

Project Sponsor
Dr. Rui Wu
Associate Professor
East Carolina University

External Advisor
Ben Gallagher
Security Analyst
University of Nevada, Reno

February 12, 2021

1. Abstract

The project team is working on developing WARE, a full stack web application designed for college robotics classes. WARE utilizes the OpenAI Gym framework, an existing open-source project, and provides a user interface to the framework's robotics environments. Users develop and test code within each environment. All processing of code is done on a back-end server, relieving the user's device from heavy computations.

This paper outlines a detailed description of the project, and indicates why it's development is significant in today's market. Furthermore, this document details how the team will manage responsibilities and goals for the project, as well as some of the legal and ethical concerns that may come up during development.

2. Project Description

This section outlines the goals and objectives of the project, as well as the unique requirements necessary for a successful outcome.

2.1 Project Goals

The primary goal of WARE is to give users an easy and accessible way to interact with the robotics environments included in the OpenAI Gym framework. A full stack web application has been developed over this last semester that lays the groundwork for the project. A back-end server was developed to handle HTTP requests, a front-end website was created for users to visit and interact with, and a database was implemented to store data on each robotics environment.

For this semester, one of the project's main goals is to improve upon the web interface and make it more responsive for devices of all sizes. Another main goal for this semester is to create a user account management system for the WARE site. This system will allow users to create an account that will store the progress of each environment that the user completes. Finally, the last goal for this semester is to publicly host the WARE website, so that it will be accessible by any device connected to the internet.

2.2 Intended Users

The WARE web application is specifically being developed for college robotics students. It is intended to be used as a tool for homework and lab assignments that involve applying machine learning concepts to robots. This web application will not only provide

a platform on which they can use to complete their assignments, but will also avoid the time-consuming and computationally intensive tasks of installing and running the OpenAI Gym framework locally.

2.3 Core Functionalities and Technologies

WARE will provide the following functionalities to the user: a live code editor with syntax highlighting, fast code compilation, a method to save the progress of the current environment, and the ability to access the site from any Internet-connected device. Bootstrap, CSS, HTML, and Javascript will be used to further develop the front-end of the website. A Javascript plugin called CodeMirror will be implemented to provide syntax highlighting in the code submission box. On the back-end, a Flask server will serve the pages for the WARE site. Code compilation is done using the OpenAI Gym framework, which outputs the results to the Flask server. MySQL is used for the database that houses the environment and user account information. The team plans to use DigitalOcean to publicly host the website using a free license provided by the GitHub Student Developer pack.

2.4 Dependability Requirements

Because offsite code compilation is one of the unique capabilities of WARE, the application will need to have high reliability in order to sustain the processing demand that is incurred by such a task. To ensure that the application remains online and accessible to users, the project team will implement appropriate performance measures on the back-end for handling multiple requests. On the issue of security, the project team will implement strong encryption and authentication measures to safeguard user passwords and data at rest and while in transit. Furthermore, the project team will ensure that all programming code that is input by a user within an environment is sanitized to defend against cross-site scripting attacks and protect the internal server from unauthorized manipulation.

3. Significance

This section defines the significance of the project in the current market, the project's potential impact on our team and society at large, as well as potential for further product development following the end of this semester.

3.1 Project Worth

The WARE project is worth pursuing as it provides students with the ability to develop and execute code in a variety of robotics environments remotely. Although the OpenAI

Gym framework is an open-source project, the installation process is time consuming and limited to use in only a handful of platforms. Additionally, running environments provided by OpenAI Gym is computationally intensive, which is problematic for devices with minimal processing power such as smartphones and tablets. Through the use of WARE, the need for students to install OpenAI software and process environments will be negated as they may execute their code remotely through a web interface, allowing for interaction with the software on devices with lower processing power or platforms with compatibility issues.

3.2 Professional Growth

Throughout the development process, our team will be working in a professional environment and will utilize the software development process to ensure the project's success. In addition to gaining experience working within a professional software engineering environment, this project will provide our team with knowledge of technologies such as the Flask web framework and the Bootstrap CSS framework. Overall, this project will help our team establish a professional software product from the beginning of its software development lifecycle to its end, which will be greatly beneficial to our professional development and provide us with practical experience for future employment scenarios.

3.3 Innovative Characteristics

Similar applications exist in the marketplace that allow students to practice their programming skills, such as HackerRank, Exercism, and Coderbyte. However, WARE has unique features that provide students with a comprehensive platform to practice robotics concepts. Innovative characteristics of our team's project include a class management system that allows instructors to provide their students with a set of environments to develop solutions for, providing the functionality of the OpenAI Gym software remotely such that users may utilize the software without installing it locally, and access to the WARE website using a mobile device. Additionally, our team plans to allow for instructors to view solutions for an environment posted by their students.

3.4 Similar Applications

Some currently available products such as Coderbyte allow for users to view solutions for challenges posted by other users; however, WARE will allow instructors to structure their student's experience and view student submissions while preventing students from viewing their colleague's work. Exercism provides a similar functionality to WARE in that users are paired with a mentor to discuss their submissions; however, WARE provides instructors with the ability to choose which environments their students can access, and

unlike Exercism, provides these services directly from the web interface without local software.

3.5 Market Potential

The main market for this application would be instructors from academic institutions interested in providing their students with an interactive method for practicing robotics concepts. Instructors will be able to set up classes for their students using the web interface provided, including the set of environments their students are allowed to work on. WARE has been designed to be intuitive to use, so that instructors are able to quickly and efficiently set up learning environments that can be utilized by their students. Overall, WARE will be incredibly useful for instructors of robotics courses, as it is easy to set up, and provides instructors with a hands-on application that their students can use to practice robotics concepts.

3.6 Potential for Further Development

Future development for the WARE project may include the addition of new environments, an API for advanced user interaction with the back end server, and further enhancements to the appearance of the website itself. Additional environments and the creation of an API would provide users with more opportunities to practice robotics principles, and continued refinement of the website's appearance would ensure that users are able to intuitively interact with the product over time.

3.7 Social and Environmental Impact

Although WARE is intended primarily for use by robotics students, this application will help to improve society by providing students with a tool for practicing robotics concepts. Through advancing the knowledge of students in the field of robotics, WARE would indirectly benefit society at large as students would be more capable of applying theoretical robotics principles learned from in-class activities to the real world.

Additionally, WARE would provide students with the ability to interact with the OpenAI Gym software, which is computationally intensive and battery life degrading on mobile devices. Furthermore, students utilizing platforms otherwise incompatible with OpenAI Gym would have the opportunity to engage with the framework without purchasing specific hardware and software capable of running it locally.

4. Legal and Ethical Aspects

It doesn't appear that the project will face any sort of legal conflicts or challenges such as those involving copyright or trademark infringements. However, there is a licensing issue that is worth noting. The OpenAI Gym robotics environments require the use of MuJoCo, a proprietary software package. The use of MuJoCo will not affect the project this semester as students are able to get a one-year license for free. However, this will affect our project sponsor as currently they would have to pay $500 per year for a MuJoCo license in order to operate the application. This is something the team will need to bring up and discuss in the next meeting.

To begin discussing professional standards, it's important to look at it from a cyber security perspective. As with any application containing information stored in a database, we ethically and morally must use the latest security standards and protocols to better defend and protect user data. We will be utilizing strong encryption standards as well as password salts to help prevent account credentials from being exposed. This information is highly sensitive and there is a high likelihood that users are reusing the same credentials across multiple websites. We will also be spending a lot of time making sure there are no vulnerabilities that can give hackers an entry point to our application or the information within it. Lastly, as our user's grades will depend on this application, the team needs to spend a lot of time ensuring that the application runs as smoothly and bug free as possible to avoid negatively affecting something as important as grades. Essentially, we cannot take short cuts on security and overall quality as that is not in the best interest of both our users and the ACM code of ethics.

5. Changes and Progress since the Initial Project Concept

Progress has been moving very smoothly since the project's last major milestone in December 2020. One major accomplishment for the project was learning how to use Flask to develop the back-end of the website. Implementing the Flask server software has allowed the team to develop a lightweight back-end server that renders the HTML templates for the front-end website and sends requests to the OpenAI Gym framework to compile code. Additionally, the Flask server gives the output of the code compiled in the form of both a video of the robot and a terminal output. Another major accomplishment was developing the front-end website using the Bootstrap framework.

One change that the team has decided to make for the short-term is developing an account management system for all users in general. Previously, the team planned to implement two types of user accounts: student accounts and instructor accounts. In light

of the other goals that the team would like to accomplish for the project, the team has currently decided to develop a basic account setup mechanism to allow users to save their progress in each environment. Creating a distinction between students and instructors will be a future goal of the project as it is further developed in the semester.

6. Project Responsibilities

With the combined effort of the project team, the project currently has a few HTML templates, a functional backend, and the beginnings of a database in place. Currently the main subsystems in development are the front end, back end, database, a text editor (CodeMirror), an account management system, a method of hosting the website, and a system to process and run submitted code. Continuing forward, the project team has divided up responsibilities for these subsystems amongst the team members or in conjunction with multiple team members.

Zach is tasked with working on the front end of the project – creating HTML templates and polishing up prior templates that the team made last semester. Templates include the home page, environment selection page, individual environment page, as well as user homepages. Zach will also be tasked with hosting the website. Herman will be in charge of implementing CodeMirror – a text editor that will give users a professional and high-quality interface to program in Python as they work through the environment's specific goals and objectives. Zach will also help implement the CodeMirror plugin. Sean will oversee the creation and design of the database. The database will store information regarding user accounts such as credentials and class ID codes as well as the user's progress in environments. Sean will also be responsible for submission processing and making it operate securely and reliably. Lastly, Ryan will be responsible for the implementation of the account management system. This will be done using a Flask plugin called Flask-Login. This will allow users to create their own accounts and join a class administered by their instructors. Ryan will also help Sean implement the submission processing mechanism.

7. Project Monitoring and Risks

7.1 Progress Monitoring Strategy

We will monitor the project's status by identifying the most important items that need to be done and assigning them to team members. For instance, one main priority is to make a database for student login information. We assigned this element to two team members to sufficiently complete the task and report back to the rest of the team if they need additional assistance.

Additionally, it is important for the project team to remain in contact with each other during the duration of the project development. During one of the team's previous meetings, it was decided to meet weekly every Friday at 3pm to update the team on the progress that has been made and to discuss the next phases of project development.

7.2 Project Risks and Mitigation Strategies

1. *Bad actor submits malicious code*
    - Mitigation Strategy: Implement technique to sanitize code and ensure users do not submit malicious code.
2. *User account information compromised*
    - Mitigation Strategy: Develop appropriate authentication procedures at login; encrypt user credentials when stored in the database; store user passwords using a salted hash.
3. *Difficulty understanding OpenAI documentation*
    - Mitigation Strategy: Reach out to Robotic Professors with more information on the subject and discuss with them strategies for tackling robotics coding and implementation.
4. *Difficulty understanding Flask documentation*
    - Mitigation Strategy: Following online Flask Pallets guide and documentation to gain a better grasp of how the software works.
5. *Project schedule is not clearly defined*
    - Mitigation Strategy: The team will have weekly meetings to discuss project progress.
6. *Difficulty processing simultaneous user submissions*
    - Mitigation Strategy: User submissions will be stored in unique files with the latest submissions linked to their account ID; Users are only allowed to make one submission at a time.
7. *Difficulty hosting project website*
    - Mitigation Strategy: Utilizing GitHub student developer pack to host the website and test it out.

8. *Users have difficulty navigating project website*
    - Mitigation Strategy: Use good UI design frameworks and appropriate font styling; apply adequate padding and spacing where needed; add basic accessibility features.

7.3 Risk Register

Risk Register											
Risk Id	Risks	Current Risk			Status	Owner	Raised	Mitigation strategies	Residual Risk		
		Likelihood	Impact	Severity					Likelihood	Impact	Severity
Category 1: Malicious actors											
1	Bad actor submits malicious code to webserver	4	4	16	Open	Team	February 5th	Implement technique to sanitize code and ensure users do not submit malicious code.	3	3	9
2	User account information compromised	3	4	12	Open	Team	February 5th	Develop appropriate authentication procedures at login; encrypt user credentials when stored in the database; store user passwords using a salted hash.	2	2	4
Category 2: Team development											
3	Diffculty understanding OpenAI documentation	2	3	6	Open	Team	February 5th	Reach out to Robotic Professors with more information on the subject and discuss with them strategies for tackling robotics coding and implementation.	2	2	4
4	Diffculty understanding Flask documentation	2	2	4	Open	Team	February 5th	Following online Flask Pallets guide and documentation to gain a better grasp of how the software works.	1	2	2
5	Project schedule is not clearly defined	2	2	4	Open	Team	February 5th	The team will have weekly meetings to discuss project progress.	1	1	1
Category 3: Product Usability and Deployment											
6	Difficulty processing simultaneous user submissions	4	4	16	Open	Team	February 5th	User submissions will be stored in unique files with the latest submissions linked to their account ID; Users are only allowed to make one submission at a time.	2	4	8
7	Difficulty Hosting project website	1	4	4	Open	Team	February 5th	Utilizing GitHub student developer pack to host the website and test it out.	1	4	4
8	Users have difficulty navigating project website	2	4	8	Open	Team	February 5th	Use good UI design frameworks and appropriate font styling; apply adequate padding and spacing where needed; add basic accessibility features.	1	3	3

8. Contributions of Team Members

This section details the time allocated and sections completed by each team member in creating this document.

Zachery Wiles

Time spent: 4.5 hours
Contributions: Drafted the Abstract and Project Description, contributed to the risk management table, added content to the Changes and Progress Made section, edited and provided feedback on other sections done by the team.

Sean Griffith

Time spent: 4.5 hours
Contributions: Created the cover page, wrote the Significance section, researched and listed references, contributed to the risk management table, edited and provided feedback on other sections done by the team.

Herman Hira

Time spent: 2.5 hours
Contributions: Wrote the Changes and Progress section and Project Monitoring and Risks section, contributed to the risk management table, edited and provided feedback on other sections done by the team.

Ryan Lunt

Time spent: 3.5
Contributions: Drafted the Project Responsibilities section and Legal and Ethical Aspects section, contributed to the risk management table, edited and provided feedback on other sections done by the team.

9. References

HackerRank, https://www.hackerrank.com/. Accessed 10 February 2021.

Exercism, https://exercism.io/. Accessed 10 February 2021.

Coderbyte, https://coderbyte.com/. Accessed 10 February 2021

ACM, https://ethics.acm.org/. Accessed 11 February 2021.