

Team Name: TechNoobs
Project Name: ECEN Capstone

Team Members :

1. Souryendu Das (Product Owner)
2. Vaidhyesh Padma Sundar (Scrum Master)
3. Swarnabha Roy
4. Aditya Thagarathi Arun
5. Sambandh Bhusan Dhal
6. Satya Sreenadh Meesala

Final Report

1. The project

The project is the management system for ECEN Capstone 403/404 course. The website features an interactive GUI with various functionalities. It allows students and admins to login using their NETID and password. The login system enables the application to have two different views: one being the student view and the other being the admin/TA view. The students have to upload their resumes and form teams. There is a view for the sponsors as Based on this, a Machine Learning algorithm is able to match them to relevant project recommendations. This ML model considers various parameters like GPA, resume skills, core competencies and project requirements to make these recommendations. The admins/TAs of this course can view these recommendations and assign them to the students accordingly.

Another aspect of the project is inventory management. The admin/TA can also check/edit/issue parts/items from the parts inventory catalogue, while students only have the option to request parts/items. The admin also has an option to enter sponsored projects from external/internal sources into the system. Overall, the system involves the use of multiple databases and forms to collect and store the information, and an interactive GUI ties it all together.

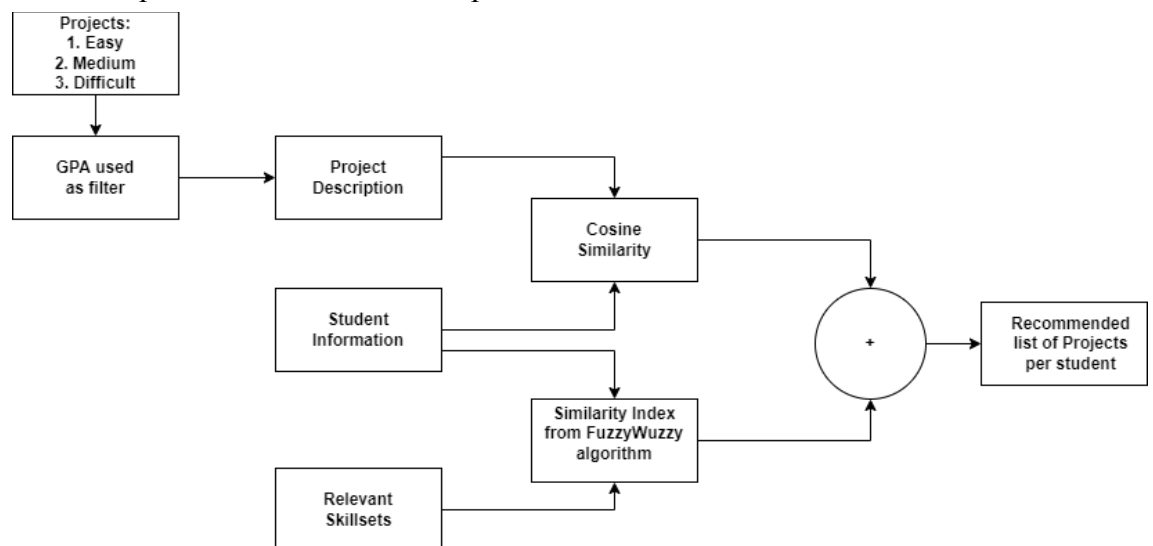
2. User Stories

1. User story 1: Set up the CI part of the pipeline (3 points)
 - a. Description: The developers can immediately produce consumable images once their code is committed, the Continuous Integration part of the pipeline was set up and the built images were pushed on to docker hub..
 - b. Implementation status : Complete
2. User story 2: Set up the CD part of the pipeline (3 points)

- a. Description: The developers can generate a working deployment on AWS BeanStalk immediately after they commit the code.
 - b. Implementation Status: Complete
3. User story 3: Instructor login and authentication using TAMU Netid (3 points)
- a. Description: Admin/faculty can log in to the ECEN-Capstone project manager using TAMU NETID, the system identifies the email being used to login and opens admin mode if the email is whitelisted as administrator. The authentication is done on TAMU's Azure Active Directory using Oauth protocol.
 - b. Implementation Status: Complete

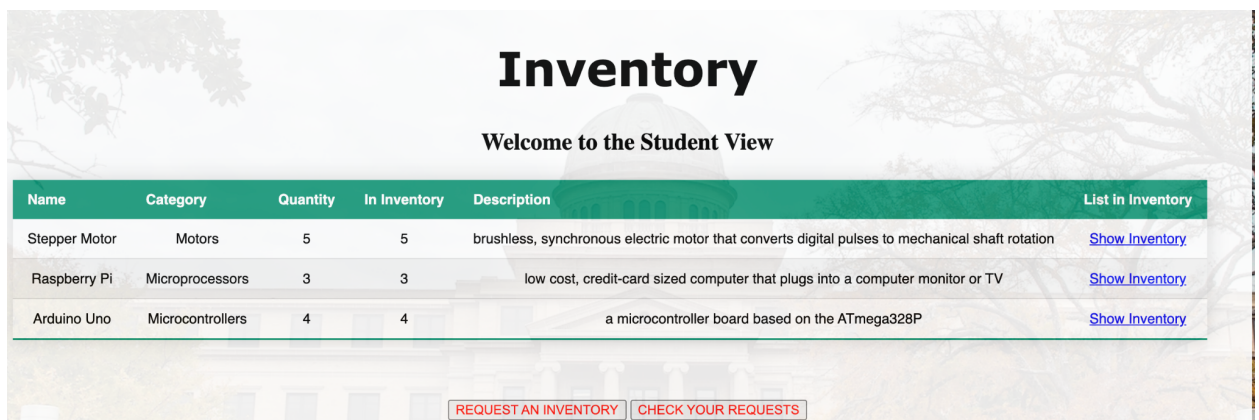
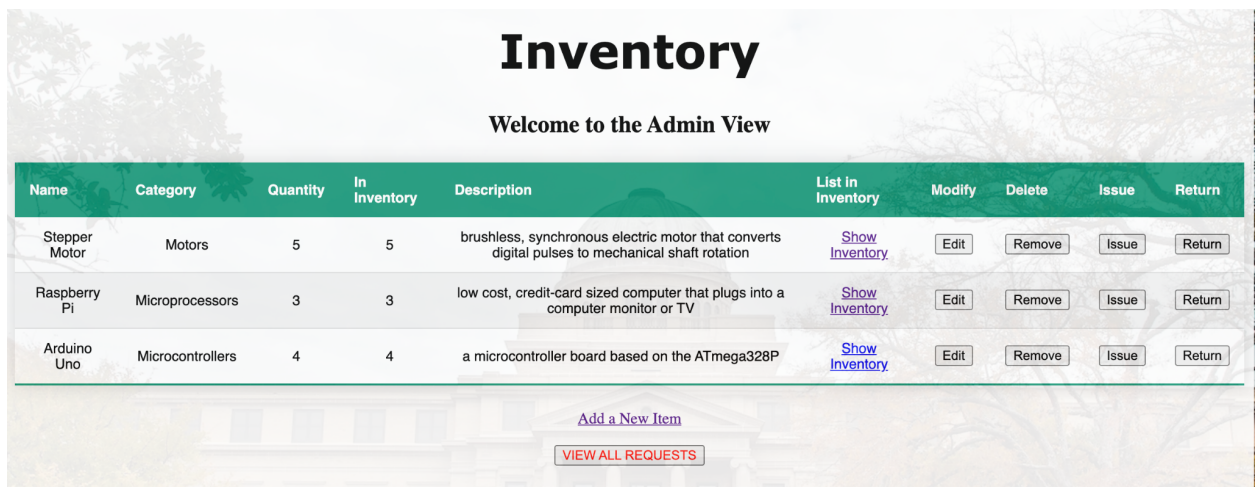
4. User story 3: Student login and authentication using TAMU Netid (3 points)
- a. Description: Students can log in to the ECEN-Capstone project manager using TAMU NETID, the system identifies the email as being a non-admin and opens the application in student mode. The authentication is done on TAMU's Azure Active Directory using the Oauth protocol.
 - b. Implementation Status: Complete
5. User story 5: Database creation for project upload by sponsors (3 points)

- a. Description: a database was created that would dynamically update the database with the projects put up by sponsors so that the project sponsors can upload multiple projects (before session starts) and all unique entries are appended to the database
 - b. Implementation Status: Complete
6. User story 6: Database creation for resume upload by students (3 points)
 - a. Description: The database was dynamically updated with the student's resume so that the students can have multiple uploads of their resume but only latest one is updated in the database (before session starts)
 - b. Implementation Status: Complete
7. User story 7: Project list recommendation to admins based on GPA, internships, and projects they have worked on (3 points)
 - a. Description: Based on the level of difficulty of projects i.e. Easy, Medium and Difficult, GPAs are used as filters to allocate the projects to the students. The keywords from the project description as entered by the sponsor and the student information from the students' resumes are extracted using Gensim summarization algorithm and a cosine similarity is found between them. Then, the FuzzyWuzzy algorithm is used to get the match percentage between relevant skill sets as entered by the user and the student resume information. After this, an aggregate of the match percentage and the cosine similarity is calculated and the final recommendation list of projects is displayed per user.
 - b. Implementation Status: Complete



8. User story 8: Enable inter container communication between multiple repositories (3 points)

- a. Description: Two different components are able to communicate with each other through a docker compose yaml file with the appropriate configurations to establish the communication channel
 - b. Implementation Status: Complete
9. User story 9: Create Database for Inventory Management (3 points)
 - a. Description: Inventory should have a database in which new items would be marked and added, edited, deleted and issued to students
 - b. Implementation Status: Complete
10. User story 10: Inventory should have a student and admin view (3 points)
 - a. Description: Student view should be able to just view the inventory and make requests for issuing parts. They should also be able to see whatever requests they have made. The admin view should enable the admin to view, edit, delete and add items to inventory and also monitor user issue requests and take action upon them
 - b. Implementation Status: Complete



11. User story 11: Implement the UI screens for interaction with backend and machine learning model (3 points)

- a. Description: The website basically is an implementation of existing ecencapstone.com website for ECE students in TAMU. The screens have to be developed for interaction with both student and admin of the website (based on the login) and then handle different data displays for each screen. Also a back to top button has been implemented.
 - b. Implementation Status: Complete
- 12. User Story 12: Create forms for Sponsors and Students to submit their details and theme the website. (3 points)
 - a. Description: There has to be a google form (I had used Google since it is recommended by the sponsor for using the Google Analytics feature) for sponsors and students that contains the details of student and sponsor. In addition to this, the entire application has to be brought in one theme.
 - b. Implementation Status: Complete

3. Team roles

Souryendu Das (Product Owner)

Vaidhyesh Padma Sundar (Scrum Master)

There were no changes to roles during the project.

4. Iteration summaries

- Iteration 0 (9 points)
 - Set up Github repository
 - Set up a Pivotal tracker
 - Set up Slack
- Iteration 1 (9 points)
 - Research and logistics of TAMU login and authentication
 - TAMU IT department has been contacted regarding:
 - Logistics to use the TAMU CAS.
 - Requested information on how to get course registration information from NETID authentication to allow only students registered to the ECEN capstone course to login to this project manager system.
 - cas.tamu.edu will not respond to any authentication requests till a form has been filled to register the application URL to the service registry.
 - Database creation for resume upload by students in progress
 - Database creation for project upload by sponsors in progress
 - Database creation for inventory management with a simple CRUD application based on sqlite

- Iteration 2 (9 points)
 - Set up the CI part of the pipeline
 - Enable inter container communication between multiple repositories
 - Advanced UI features to improve the readability for users in mobile, desktop and tablet modes
 - Testing of Inventory Management with more use cases in BDD and TDD
 - Login and authentication using TAMU NETID (in progress)
 - Upon contacting the IT department we were advised against using the initially planned CAS system for authentication.
 - Instead, we were advised to leverage the university's Azure AD environment to perform our authentication using OIDC/OAuth 2.0 protocols.
 - An application called "ECEN capstone manager" has been registered
 - Currently implementing the protocol in rails.
 - Authentication pass through **issue/problem** with Azure Active Directory, IT department contacted about the issue
- Iteration 3 (9 points)
 - Database creation for project upload by students and sponsors finalised and completed
 - Setup CD part of the pipeline
 - Testing of recommendation system: Recommend a list is 5 projects to the students based on GPA, Internships, and projects they have worked on
 - Creation of student view and admin view, basic functionality implementation
 - Login and authentication using TAMU NETID (in progress)
 - Able to log in, waiting for further permissions for application redirection after authentication.
- Final Iteration/Iteration 4 (18 points)
 - Login and authentication using TAMU NetID implementation completed
 - Recommend a list is 5 projects to the students based on GPA, Internships, and projects they have worked on completed
 - UI beautification
 - ML, inventory management integration with frontend
 - Inventory management distinguishes views based on session variables. Auto population of user unique fields and adding security to information inside the database.
 - Adding checks to validate and disallow cross site navigation to unwanted URLs in the website without following the proper flow, i.e., if an user just

types in the URL of a certain page in the website, without following the proper flow, they are denied access to the page

5. Customer meeting dates, feedback, and software/stories demoed.

Meeting once every 2 weeks with the customer:

- November 5th 2021 (3pm - 4pm)
 - Discussed all project details and workings of the AI.
 - **Feedback from customer**
 - Include a marker to check how many items were checked out against inventory and were checked out against which user
 - Include user notifications and emails if they are nearby due date of returning items to inventory
 - Use google form for populating the database
- November 19th 2021 (3pm - 4pm)
 - Demoed barebones website on local host as there was a problem with deployment(barebones inventory system and frontend).
 - **Feedback and queries from the customer**
 - Student resume should be a file vs URL. It is likely they do not have it publicly available?
 - The proposed 5 projects should be only to the Instructor, not to the student.
 - Inventory manager is only visible to TA from the lab?
- December 1st, 2021 (3pm - 4pm)
 - Demoed login system, discussed fine tuning of the ML model, discussed how students would be mapped to projects.
 - **Feedback from customer**
 - The website should return a list of at most five projects that are common to all the members of the team.
 - If no project comes in common, the instructor will assign the projects manually

6. Explain your BDD/TDD process, and any benefits/problems from it.

The BDD and TDD process started for the Inventory Management controller and the welcome controller was implemented in Rails. The welcome controller talks about the aspects related to Welcome Page, menu and the CSS involved with it. The BDD/TDD helped us to check navigation amongst the pages. The items controller worked with the

inventory management database on a CRUD level and BDD/TDD helped us test all the aspects of Create, Read, Update, Delete, duplicacy and security.

7. Discuss your configuration management approach. Did you need to do any spikes?
How many branches and releases did you have?

Spikes were used for the authentication part of the project. There was a spike put up to research the TAMU central authentication system to gauge the difficulty of this endeavour before there was a shift to the use of the university's Azure AD. Even after there was a shift to Azure AD an approach to using Devise was tried out.

The second spike in this case was the research regarding the NLP model. There was a spike put up to research on how to devise the similarity indices using the NLP models for accurate allocation of projects to the sponsors. Different algorithms were researched on and implemented to find out which would best serve the purpose of resume matching with projects.

We had three branches: dev, release, inventory and three releases in all.

8. Describe any issues you had using AWS Cloud9 and GitHub and other tools.

The free amazon cloud9 tier instance allowed us to store 1gb of data which was insufficient for our docker system, so we moved to developing our app locally. There were no issues with GitHub.

9. Describe the other tools/GEMs you used, such as CodeClimate, or SimpleCov, and their benefits.

Gems:

Httparty: this gem allowed to make rest calls

Omniauth-oauth2: This allowed us to leverage the university's Azure AD

10. Links to your Pivotal Tracker, public GitHub repo, and Heroku deployment, as appropriate. Make sure these are up-to-date.

Github Repo: <https://github.com/TAMU-ECEN-CAPSTONE-Manager>

Pivotal tracker link: <https://www.pivotaltracker.com/n/projects/2534348>

Link to deployment: <https://tamu.ecencapstone.xyz/>

11. Links to your poster video and demo video.

Demo video: <https://youtu.be/52I6Vk9ZeWE>

Poster video: <https://youtu.be/SnEECN7ykiI>