|  |  |  |
| --- | --- | --- |
| *Course Title and Term* | CEN 4010 Principles of Software Engineering, Spring 2018 | |
| *Document Name* | Milestone 1 Project Proposal and High-level description | |
| *Team Name/Project Name* | Group 7 - Web Store for Electronic Lab at College of Engineering | |
| *Team Number* | Group 7 | |
| *Names of Students* | Ayah Elshaikh – aelshaikh2015@fau.edu  M. Kaan Tasbas  Taylor Tanksley  Brian Beltran | |
| *Date* | 27 March 2018 | |
| *History Table* | **Date** | **Activity** |
|  | 2/15/2018 | Edited executive summary, adding more information from notes taken after meeting Perry |
| 2/17/2018 | Completed competitive analysis and high-level functional requirements |
| 2/18/2018 | Completed data definition and edited non-functional requirements |
| 2/19/2018 | Completed overview, scenarios, and use cases |
| 3/27/2018 | Performed proper revisions to M1 |

# Executive Summary

## Product Name: Perry’s Parts Pavilion

Perry’ Parts Pavilion is a reinvented version of the CEECS Lab Services request form. It allows the user to navigate through parts, sorted by a variety of categories, and makes it easier for them to find what they’re looking for. The webstore also allows the user to reserve the parts they need and save them in a cart. If an item is currently unavailable, the webstore will allow the user to place a request for it and will notify them once it is available. If the user needs to rent an item, a check-out and check-in transaction will be completed. The condition of the item will be evaluated before and after it’s checked out. Once the user completes their request, they will be directed to a checkout page and given a verification number.

To access the webstore, the user must create an account, then must use his or her FAU username and password to log in. Upon login, the user will be directed to the OIT (office of information technology) verification page and redirected back to the webstore. Those who are not CEECS students will need to be granted permission to access the webstore.

One of the webstore’s key advantages is the ability to search and locate needed parts and read a description about the parts. Certain keywords will be assigned to the products, and the search results will be based on that keyword. The quantity of each item will also be displayed for the user. If an item is not available, the user may upload a picture of the part and request it to be ordered. The items displayed will have the product name, product description, and quantity.

# Competitive Analysis

|  |  |
| --- | --- |
| Perry’s Parts Pavilion Features | Competitor Features |
| Equipment Loan/Repair | Equipment Loan/Repair |
| File Upload | File Upload |
| Availability Status | Priority Level |
| Invoice with Part ID Number | Confirmation Email |
| Part Search |  |
| Part Description |  |
| Pictures of Items |  |

## Advantages of Perry’s Parts Pavilion

Perry’s Parts Pavilion allows the user to checkout multiple items, rather than one at a time. It also provides the user with the availability status of an item. If an item is currently unavailable, the user is able to request it and will be notified when it is once again available. The webstore also to upload files for specific services, such as 3D printing and PCB milling. It also provides them with a reference number for their requests, which they can refer to if there is any mix-up. Users may search through the items using the “Parts” section of the webstore to locate the items they need. The webstore also provides the user with the ability to search for an item through the search bar and will provide a short description for each part. The description will include a keyword which allows the item to appear in the search results. The images that will be shown for the items will allow the users to easily find the parts they need when they don’t know the specific name or part number.

|  |  |
| --- | --- |
| Name | Usage |
| partID | Part ID variable – allows for the storage of the unique ID code for the parts |
| partName | Name variable – allows for the storage of the name of the specific parts |
| partCategory | Category variable – allows for the storage of the category in which each part belongs to |
| partDescription | Description variable – holds the description of each part |
| serviceID | Service ID variable – allows for the storage of a unique ID number for each service |
| ticketID | Ticket ID variable – holds the number of the ticket that user fills out for part requests |
| transactionID | Transaction variable – holds a unique number for the transaction that the user completes |
| inventoryStatus | Inventory Status variable – stores the availability status of the parts |
| amountLeft | Amount variable – holds the number of available parts when it is less than 10 items |
| partPrice | Prince variable – used to show the prices of the items listed in the webstore |
| permissionType | Permissions variable – controls the type of access different users get (staff, students, etc.) |
| studentID | ID variable – stores the user’s Z-number |
| studentName | Name variable – stores the name of the users of the webstore |
| classID | Class variable – stores the code of the class that the user requires a part for |

# Data Definition

|  |  |
| --- | --- |
| Name | Usage |
| collegeID | College variable – holds the name of the college the user belongs to |
| studentFirstName | First name variable – stores the first name of the student/account owner |
| studentLastName | Last name variable – stores the last name of the student/account owner |
| phoneNumber | Phone number variable – stores the student/account owner’s phone number |
| numberofLoans | Loan amount variable – holds the number of loans student/account has |
| numberofOrders | Order amount variable – holds the number of orders the student/account has |
| totalPrice | Total price variable – holds the total price of order made |
| confirmationNumber | Confirmation order variable – stores the confirmation number for successful order made |
| studentEmail | Email variable – holds the student/account email |
| studentPassword | Password variable – holds the student/account password |

# Overview, Scenarios, and Use Cases

## Purchase Parts Scenario

Elaine would like to build a temperature sensor system and for her to do so she needs to purchase certain parts. For instance, she needs an Arduino, temperature/humidifier sensor, bread board, female and male wires. For her to purchase this at Perry’s Parts Pavilion, she has to create an account, so she is able to sign in and access the webstore. Once she is on the webpage she should be able to see the sign-up tab, click on it and it will navigate her to a page where she is able to input her first and last name and the school she attends. Students that attend Florida Atlantic University must supply their owl card. After Elaine sets up her account, she is now able to purchase the parts. She selects the parts that she wants to purchase and add them to her cart for checkout, for any part that she does not know the detailed specs about she is able to upload a picture of the part for further assistance. However, before Elaine can checkout, she must specify whether the order is for a personal project or class project. Students can gain access immediately after purchase by going to lab and swiping their owl card. If an ordered part is out of stock, then Elaine will be notified when the item is available.

## Checkout Equipment Scenario

Kramer would like to solder some parts together and for him to do so he needs to checkout some soldering equipment from the school. The items he would need includes a soldering iron, solder, and solder suction tool. For him to checkout this equipment through Perry’s Parts Pavilion, he has to create an account, so he is able to sign in and access the webstore, if not already done so. If Kramer has not created an account on the Webstore, then once he is on the webpage he should be able to see the sign-up tab, click on it and it will navigate him to a page where he is able to input his first and last name and the school he attends. Students that attend Florida Atlantic University must supply their owl card. After Kramer sets up his account, he is now able to find equipment to checkout. He selects from our inventory the items that he wants to checkout and add them to his cart for checkout. Students can gain access immediately after checkout by going to lab and swiping their owl card. Kramer must return the checkout items back in by the end of the day. The original condition of the items but be kept by Kramer upon return. He will experience financial penalties if items are return damage.

## Report Equipment Problems Scenario

George would like to report a problem that he experienced when using equipment that he has checked out through our Perry’s Parts Pavilion. In order to do this, George must sign into his account that he has with our webstore. From the home page, George should select the fill out a ticket option from the navigation bar. He should fill out the fields that are asked for and submit the ticket. After submitted the ticket, George should return to the lab and wait to get assisted lab assistant.

# High-Level Functional Requirements

* The user will be able to search for specific parts through the webstore (1)
  + The user may enter the name of an item in the search box, or search through all the items by clicking on “Parts”
* The user may search through specific services offered (1)
  + By clicking the “Services” button, the user may request from the following services: 3D printing, PCB milling, laser-cutting troubleshooting, equipment loan, part request
* The user will be able to upload images for part requests and services (1)
  + This can be done by clicking on “Services”, and the user will be given an option to upload a file for the desired service (3D printing, PCB milling, etc.)
* Every item in the store will be allocated a unique ID number (1)
* The system will allow users to create a new account, and send a verification email (2)
  + The user will sign up with first and last name, z-number and phone number, and must specify the college they are a part of
* The user may add all the parts to a cart (2)
  + The cart will store the items and their quantity, and gives the users the option to checkout
* Upon checkout, the user will receive an invoice (2)
  + The invoice will include the confirmation number, names and quantity of items ordered, and the date and time of the checkout
* Images will be shown for parts (3)
  + If the user does not know the specific name, the images will help in their search

# Non-Functional Requirements

* Performance: Quick page load (within 2 seconds); does not slow down significantly when there is a lot of traffic (over 100 students using the webstore at once)
* Usability: The webstore’s functionalities can be learned in under 10 minutes
* Accessibility: The website does not translate to other languages and does not have any accommodation for the visually impaired
* Expected Load: 100 students per day
* Security Requirements: CEECS, OIT
* Storage: LAMP server
* Availability: 24/7 access to qualified CEECS students, students from other colleges who have been granted access by Perry and his staff
* Supportability: system maintenance is cost-effective (within the product owner’s budget)

# High-Level System Architecture

* Back-End: Java, MySQL
* Front-End: HTML, CSS, JavaScript, Python
* Supported Browsers: Google Chrome, Internet Explorer/Microsoft Edge

# Risks for Project

1. **Schedule:** The schedule for the assignment is very tight. We may not be able to include everything that we planned for the assignment

We plan to focus on the most important aspects of the assignment first, such as including the actual inventory. Less important aspects – such as images for the items and the quantity – will be included if there is time.

1. **Team:** Since there are only four members in our group, it is difficult to do all that is necessary. It will take us a very long time to complete everything we hope to achieve with the project.

Each team member will have to take on extra work to complete all the aspects of the project. To avoid any setbacks, each team member will have to work harder and longer on the assignment, and learn any skills that team is lacking.

1. **Time Management:** We underestimate the amount of time required for certain tasks. Those tasks are therefore delayed and cause us to be stressed to finish them in time.

Tasks will be started early, no matter how simple they may seem. Each member will stick to a specific timeline and make sure their portion is completed on time.

# Team

|  |  |
| --- | --- |
| Name | Title |
| C. Perry Weinthal | Product Owner |
| Ayah Elshaikh | Front-End Lead |
| M. Kaan Tasbas | Back-End Lead |
| Taylor Tanksley | Scrum Master |
| Brian Beltran | GitHub Master |