Course: CEN 4010 (Principles of Software Engineering)

Semester: Spring 2018

Milestone 3: More Detailed Requirements, Architecture and a Vertical Software Prototype

Team name: Five Bros Chilling in a Hot Tub (Group 3)

Project name: Florida Atlantic University Original Web eLectronics Store (FAUOWLS)

Team:

* Joshua Cordes, team lead, jcordes@fau.edu
* Martin Daniel Sanchez
* Brett Marks
* Alvaro Feola
* William Young

Date: 26 March 2018

|  |  |
| --- | --- |
| History Table | |
| 19 Feb. 2018 | Initial submission of Milestone 1 |
| 22 March 2018 | Revisions of Milestone 1 added to Milestone 3 |
| 26 March 2018 | Initial submission of Milestone 3 |
|  |  |

2

From Room 205 of Florida Atlantic University’s (FAU) Engineering East building, Perry Weinthal sells or leases various electronic parts and equipment, as well as manufactured pieces (such as those from a 3D printer) by request. To facilitate inventory-tracking, we are producing the Florida Atlantic University Original Web eLectronics Store (FAUOWLS), where students will be able to search for and purchase items they seek, and Perry and his team will be able to keep track of all equipment and parts as they come in and go out. Many web stores, such as Newark element14 (<http://www.newark.com>) already exist that sell electronic equipment, and Perry often purchases equipment and parts from such sites, in addition to purchasing from more general-purpose web stores like eBay (<https://www.ebay.com>).

The uniqueness of the FAUOWLS is that it would primarily target students at FAU either currently taking or who have taken courses in computer science, computer engineering, or electrical engineering, who would be able to purchase at no cost many electronic parts that would otherwise cost some non-zero amount, as well as lease or rent more complex equipment for free - equipment that would otherwise be prohibitively expensive for the average university student. FAUOWLS will also allow those on Perry’s team to locate equipment in various laboratories upon request, saving them the trouble of either having to memorize the location of every part or spend excessive time rummaging through drawers to find what a student seeks. Faculty, additionally, will be able to purchase, rent, or lease equipment or parts, at little to no cost, while FAUOWLS will allow Perry’s team to track all such prices in one central database.

Ultimately, FAUOWLS would allow Perry and his team to be more productive, and allow students and faculty needing parts and equipment to have a clear, easily accessible system for finding and obtaining what they’re looking for to excel in their courses - or even engineer their own experiments.

3. Competitive Analysis

|  |  |
| --- | --- |
| **Competitors** | **FAUOWLS** |
| Multiple Ways to Pay | No Money Exchange Needed |
| Algorithms for Personal Recommendations | Focus on Optimizing Search/Sort Algorithms |
| Focus on Advertising | Clean Presentation Focusing on Functionality |

Our focus for FAUOWLS with respect to our competitors has not changed since the previous submission. We still believe our goal should be quick functionality and clean presentation over advertisement-filled pages and personal recommendations based on user profiles. If recommendations are added later, they will be focused around courses that the user is taking. Since we are not charging users any sort of currency, there is no need for advertisements within the site, unless it is a kit or service that is ordered frequently, in which we could add a quick link on the front page. In regards to the former point, our focus being elsewhere other than advertising allows us freedom from falling in line with our competitor sites. However, we still plan to utilize the organizational conventions that other web stores use, such as categorized inventory and navigational menus. This will allow users to find what they need swiftly, along with keeping different pages of the store clutter-free. Ultimately, the store will compete with search and load times of bigger competitors, along with the ease of utility that is found among the best sites.

4

Data definition:

**Admin** – Administrator’s account. Will be able to create/edit Staff and Client accounts, approve item requests from Clients and order items from Vendors to send to Inventory.

Admin attributes:

* Username - Username with which Admin logs in.
* Password - A single password/key used by the Admin to log in to his/her account.

**Client** – Person (FAU student, faculty et al.) ordering or renting item(s) from Inventory. May also request items from Admin.

Client attributes:

* Znumber - Primary key. Client’s FAU-assigned Z-number.
* Password – Password/key for client to log in to his/her account. Created by user.
* Major - Client’s major at FAU. User-selected.
* Courses – Client’s current courses at FAU.
* Graduation date – Client’s expected graduation date.
* Cart – Stores part number for each item added for prospective purchase/rental by Client. Where final purchase or rent function can be applied.
* Purchase and Rent History – A list of all previous purchases and rentals. Includes Part Number, Order or Rent Date, Pick-up Date, and Return Date if part was or is being rented.
* Client’s Orders – Current client pending orders.

**Engineering Lab Staff** – Account that shall be able to view Client orders, view Client account history, create Kits, add items to Inventory, and add new Vendors.

Engineering Lab Staff attributes:

* Username - Each engineering staff member will use a unique log-in created by the Admin.
* Password - A single password/key for Engineering Lab Staff to log in to his/her account.

**Inventory** – Stores product inventory available for clients to purchase/rent.

Inventory attributes:

* Part number - Primary key for Inventory item.
* Short Description – A short text description of the product in minimal detail.
* Long Description – A detailed text description of the product.
* Image – Capable of storing multiple images in specified format.
* Data Sheet - Data in an Excel spreadsheet or PDF that can be uploaded.
* Location in Lab – Specifies where Inventory item located in lab.
* Quantity – Quantity of Inventory item available in lab for purchase/rent.
* Purchase or Rent – Indicates whether Inventory item available for purchase, rent, or both.
* Barcode - Bar code associated with product.
* Purchase Price – Original price per item paid to Vendor for purchase by lab.
* Ordinary Selling Price - Price per item when sold to Client.
* Bulk Selling Price – Price per Bulk Quantity at which item is sold to Client.
* Bulk Quantity - Quantity of Inventory item for which Bulk Selling Price applies.
* Jobber Selling Price - Price per item when sold to a middle man.
* Vendor part number – The vendor’s part number.
* Part availability per client – A limit set on each part that a client can order depending on the group certification.

**Vendor** – Company from which some item in Inventory is ordered or can be ordered.

Vendor attributes:

* Name – Name of vendor company.
* Address - Physical address of the vendor.
* Account number - Each vendor will have a unique account number used as a primary key.
* Order History – List of all items with Quantity and Purchase Date ordered by Admin.

5

Overview:

The FAUOWLS will be used by both Engineering Lab Staff and FAU students and faculty as a way to distribute goods for engineering projects, as well as services, generating a record of transactions. Students will be able to identify through their FAU ID, log into their account and "purchase"/"rent" any needed parts or "hire" troubleshooting and printing services. For these services, students will need to upload different types of files. Staff will be able to keep track of real-time inventory and their location; when orders are generated, these should indicate the physical location of the products. Products and suppliers can be created and modified by the Staff. Products will have an image associated to it, as well as data sheets, which can be uploaded by clients with a pending approval from the Staff, and different prices depending on the customer classification. There will be assembled products, which will be conformed of other products, affecting the inventory. They system will offer an option to add the assembled kits to stock, where it will calculate if the necessary amount of input products is in stock, if there is no sufficient stock the system will inform what is missing.

Scenario:

Jack is a Computer Engineering student taking Logic Design, therefore skilled on computer tasks. At the beginning of the semester Jack is given a kit including two breadboards, LEDs, resistors and various other parts. Unfortunately, Jack makes a mistake and causes some LEDs to stop working. On the next class, he talks to his classmates about this problem, since he needs a white LED to finish his Lab project; they suggest that he uses FAUOWLS to order the part he needs. Jack signs in to FAUOWLS using his FAU ID and Z-number, browses for the white LED he needs and orders it at no cost. Later, he goes to room 205 on Engineering East building, where Perry or another member of Perry’s team scans Client’s OWL Card, pulls up his order, which indicates where in the Engineering Lab the white LED is located, and delivers it to Jack.

Initial Assumption: The user has successfully signed up to the FAUOWLS and the item needed is in stock.

Normal: When signing up, the user must enter his personal information and Z number. When browsing for the desired item there is a search function to find the item, which has a description, photo and data sheet to avoid confusion.

What can go wrong: User may order the wrong item since parts often look alike and user may not pay attention to data sheets.

System state on completion: After ordering, the stock of the item is reduced by the amount ordered. There is a record of the transaction on the system.

Scenario:

John is a FAU engineering student who works at the Engineering Lab on Engineering East building. As part of his routine, he logs into FAUOWLS system and checks the pending orders. He finds that a student Jack made an order for a custom-built circuit board with the necessary files attached and specific comments, which John uses to build it. Upon its completion, John sends an email to the student saying the order is ready and leaves the package ready for its collection. Jack arrives to the Engineering Lab, where his OWL card is scanned by John to pull up his order and confirm his identity. Then John gets the prepared order, hands it to Jack and marks the order as completed.

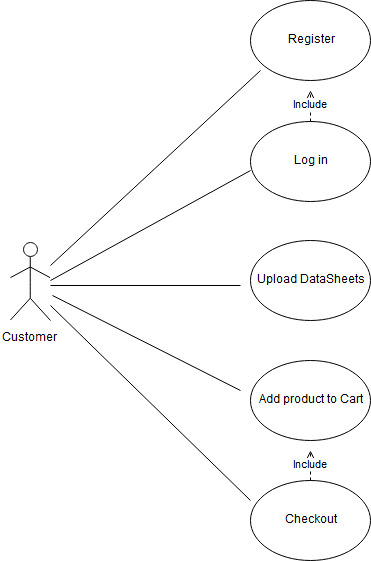
Initial Assumption: The circuit board materials and machine are in stock and available.

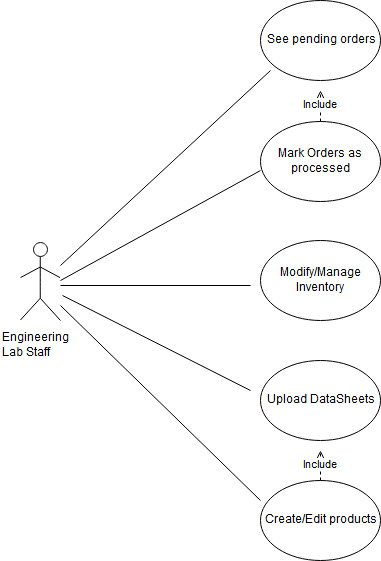
Normal: When making the order, user must include files and comments regarding the desired circuit board.

What can go wrong: User may include wrong files, wrong file type, comments or file with errors resulting in a delayed, not accepted order or a circuit board that is not desired.

System state on completion: After ordering, there is a record of the transaction on the system.

Use Cases:





6 High-level Functional Requirements

1. Account system shall allow for students or staff to make purchases using Z-number. (no real money exchanged). **1 - must have**

a. Rental/leases shall have a strict time limit.

b. Admin shall be able to track what is taken and how much to keep track and order when supplies are low.

2. Admin shall be able to apply separate price for Jobbers, those purchasing as a middle man, such as another FAU department. **2 - desired**

3. Admin shall be able to apply separate Bulk price for when client requests a large amount of an item. **2 - desired**

4. Store shall produce a live-updating spreadsheet on the website which shows past rental/lease dates, inventory etc. **1 - must have**

5. Store map shall show whoever is working exactly where an item is in EE205. 2 - desired.

6. Client shall be able to upload a file for 3d Printing, circuit board milling or laser cutting. (.jpg, .tiff, etc.) **2 - desired**

7. There shall be a list of vendors and their websites for easy access. **2 - desired**

8. Pictures for each specific item will be able to be uploaded to the webstore. **3 - opportunistic**

9. Any special orders will send an email notification to those who ordered the item that it is ready. **2 - desired**

10. There shall be a standard module implementation for barcodes. **1 - must have**

11. Students should be able to upload data sheets that can be used at the staff’s discretion. **3 - opportunistic**

12. Account should keep track of client’s purchase/rental history. **1 - must have**

13. Staff and Admin shall have privileges that Client won’t. **1 - must have**

14. Admin shall manage accounts with permissions unavailable to Lab Staff or Client. **1 - must have**

15. System shall subtract item quantity from Inventory after purchase or rental, and shall add to Inventory when rented item is returned. **2 - desired**

16. Any item request shall be submitted to FAUOWLS as a ticket. Engineering Lab Staff shall see the request immediately upon submission for order to be fulfilled. **2 - desired**

7. Non Functional Requirements:

1. To sign in to FAUOWLS, user shall connect to FAU Single Sign-on with corresponding credentials.
2. If a student orders an item for a specific course, the system shall not need to check if student is currently registered for the course.
3. Expected load is no more than 300 simultaneous users.

8. High-level system architecture and database organization

Database Organization:

|  |  |  |  |
| --- | --- | --- | --- |
| **Table name** | **Fields** | **Primary/**  **composite key** | **Foreign keys** |
| Comments | perry\_part\_num  z\_num  comment  rating  upvotes  downvotes | {perry\_part\_num, z\_num} | `Comments`.`perry\_part\_num` -> `Inventory`.`perry\_part\_num`  `Comments`.`z\_num` -> `Store\_users`.`z\_num` |
| Course\_sections | CRN  code\_letters  code\_num  title  college  department  semester  year | {CRN} | NONE |
| Course\_info\_by\_user | z\_num  CRN | {z\_num, CRN} | `Course\_info\_by\_user`.`CRN` -> `Course\_sections`.`CRN`  `Course\_info\_by\_user`.`z\_num` -> `Store\_users`.`z\_num` |
| Inventory | perry\_part\_num  short\_description  long\_description  location\_in\_lab  quantity  purchase\_or\_rent  barcode  retail\_price  retail\_price\_promo  retail\_markup  jobber\_price  jobber\_markup  bulk\_price  bulk\_markup  cost\_to\_replace  cost\_avg | {perry\_part\_num} | NONE |
| Keywords | perry\_part\_num  keyword | {perry\_part\_num,  keyword} | `Keywords`.’perry\_part\_num` -> `Inventory`.`perry\_part\_num` |
| User\_orders | order\_num  z\_num  order\_date | {order\_num} | `User\_orders`.`z\_num` -> `Store\_users`.`z\_num` |
| User\_order\_items | order\_num  z\_num  perry\_part\_num  quantity  price\_charged\_per  net\_profit  student\_bool  faculty\_bool  for\_personal\_bool  rented\_bool  return\_by  returned\_bool  delivered\_bool | {order\_num, perry\_part\_num} | `User\_order\_items`.`z\_num` -> `Store\_users`.`z\_num`  `User\_order\_items`.`perry\_part\_num` -> `Inventory`.`perry\_part\_num`  `User\_order\_items`.`order\_num` -> `User\_orders` -> `order\_num` |
| Perry\_orders | order\_num  vendor\_name  date  total\_price | {vendor\_name, order\_num} | `Perry\_orders`.`vendor\_name` -> `Vendors`.`vendor\_name` |
| Perry\_order\_items | order\_num  vendor\_part\_num  quantity  price\_per  total\_price | {vendor\_name, order\_num, vendor\_part\_num} | `Perry\_order\_items`.`vendor\_part\_num` -> `Vendor\_items`.`vendor\_part\_num`  `Perry\_order\_items`.`order\_num` -> `Perry\_orders`.`order\_num` |
| Store\_users | user\_type  first\_name  last\_name  username  address  city  state  zip\_code  z\_num  major\_1  major\_2  minor\_1  minor\_2  phone  FAU\_email  password  grad\_semester  grad\_year | {z\_num} | NONE |
| Vendors | vendor\_name  website  acct\_num  address  city  state\_province  country  phone  email  contact\_name\_last  contact\_name\_first | {vendor\_name} | NONE |
| Vendor\_items | vendor\_name  perry\_part\_num  vendor\_part\_num  price\_per | {perry\_part\_num, vendor\_name, vendor\_part\_num} | `Vendor\_items`.`vendor\_name` -> `Vendors`.`vendor\_name`  `Vendor\_items`.`perry\_part\_num` -> `Inventory`.`perry\_part\_num` |

Media storage:

* Images and other attachments (e.g., data sheets) to be stored in file folders rather than in the database.
* File types
  + Images for Inventory, service requests, data sheets: JPG, TIFF, PNG, SVG
  + 3D printing requests: STL
  + Circuit board milling service request: GBR
  + Data sheets: PDF, DOC, DOCX, JPG, TIFF, PNG, SVG

Search/filter architecture and implementation:

* All users shall query all fields for all records in databases Inventory, Keywords, and Comments.
* Admin and Staff shall query all fields for all records in databases Course\_info\_by\_user, Course\_sections, Perry\_orders, Perry\_order\_items, User\_orders, User\_order\_items, Vendors, and Vendor\_items.
* Admin and Staff shall query database Store\_users in all records in all fields but *password*.
* User shall limit query to a certain field, e.g., search by *perry\_part\_num*, which will return the specific page dedicated to Inventory item with given *perry\_part\_num*, or by Inventory *short\_description* or *long\_description*, which will return pages for items with matching search terms.
* User shall wrap search text in quotation marks to indicate exact phrases.
* For purposes of search, when user shall specify field other than `*Inventory*`.`*long\_description*` to search, database records shall be sorted by that field and use binary search.

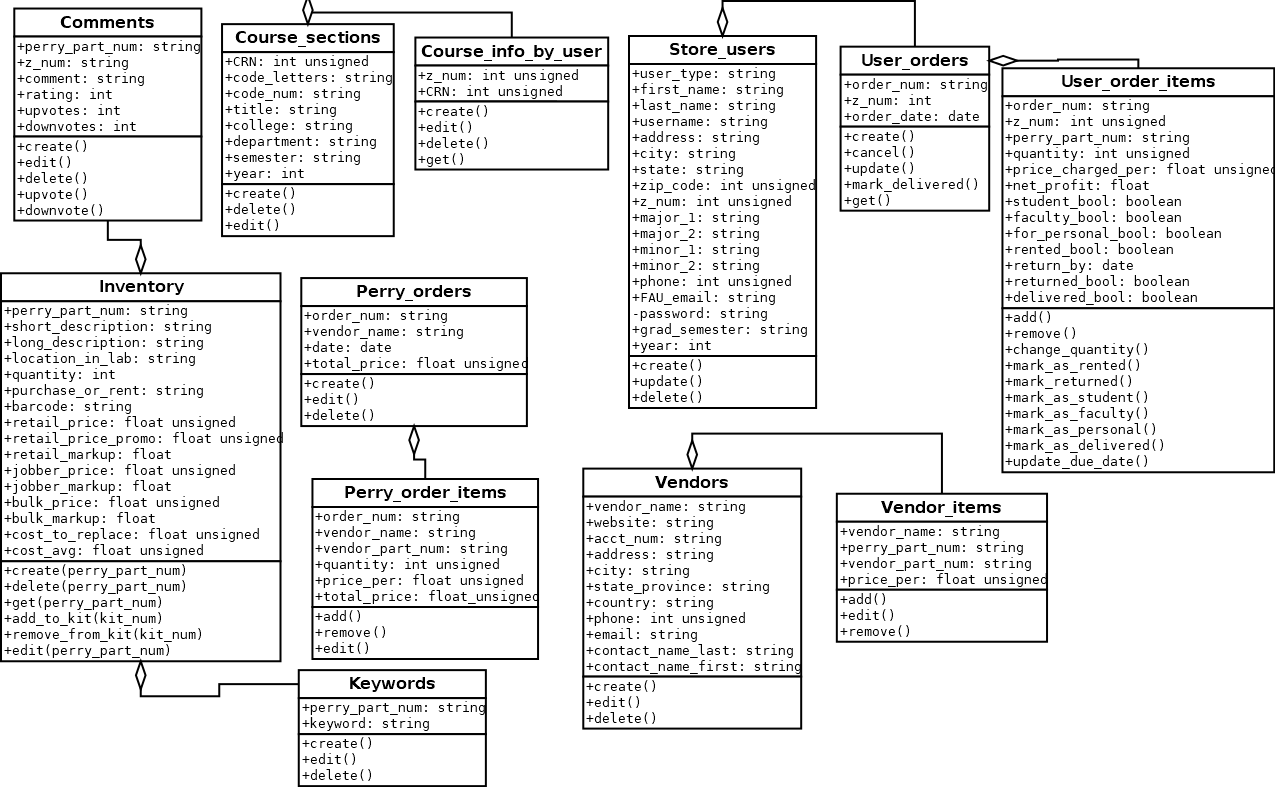
Major API’s:

* No major original API’s shall be produced for this project.

Describe any significant non-trivial algorithm or process:

* Staff and Admin shall see list of unfulfilled orders, with the ability to search by *z\_num* or *username*. List shall update live and user shall click to confirm when order has been delivered.
* Staff and Admin shall see live-updated list of service requests not completed by any Staff or Admin, with download links to any attachments submitter has included.

9. High-level UML diagrams



10. Key Risks

As it stands currently, the most relevant risks our team faces revolve around skill and schedule. While most of us have some experience in front end web development, few have experience, if any, with databases and back end development. This makes it difficult to evaluate exactly how much time we need to set up inventory and inventory management tools, and also a framework for searching the database and retrieving accurate results. The plan, at this time, is to utilize what we know collectively, in SQL and PHP, and do rapid research to find functionalities that we need and implementing them within our code as necessary.

In regards to schedule risks, there isn’t much to be done about conflicting schedules and availability. Most of our team has responsibilities outside of this project that cannot be rescheduled or avoided. We will continue as we have been with meetings and rely on each other to do their part on time and ask for help as needed. As it stands, the team has handled conflicting schedules exceptionally and will continue that trend through the duration of the project.

The final risk is in regards to legal issues. This is a point that has not been addressed yet within the group. As of March 22nd, the only issue that we have addressed that comes close to a legal or content issue is allowing users to sign in with their FAU identification, without giving us sensitive information in regards to username and passwords. We plan to solve this using PHP, sending users to an FAU login page, and giving the client an access token that grants them access to navigate our site under their account. If a user is not logged in, they can still navigate the site, but cannot complete orders and will be redirected to a login/register page if they try to access any page that is account-oriented. This might become an evolving issue that may cross into a technical concern that will be resolved when we reach that stage in development.