CEN4010 Principles of Software Engineering

Spring 2018

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

Group 9: Hardware Nanny

Alicia Mitchell (amitch41@fau.edu)

Sergio Ramirez

Bradley Rubenstein

Guimel Gonzalez Hernandez

Ryan Juraidini

Date: Monday March. 26th, 2018

|  |
| --- |
| Document Revision History Table |
| Revision 1 - 2/19/2018 |
| Revision 2 - 3/26/2018 |
|  |

**Table of Contents**

1. Executive Summary……………………………………………………………………………………………………….Page 3
2. Competitive Analysis…………………………………………………………………………………………………….Page 4
3. Data Definitions………………………….…………………………………….………………………….….……………Page 5
4. Scenarios and Use Cases……………….………….…………………………………………………………………..Page 8
5. High Level Functional Requirements...………………………………………………………………………..Page 10
6. High Level Non-functional Requirements.……………………………………...……………………….....Page 11
7. High Level System Architecture…………………………………………………………………………………..Page 15
8. Database Organization………………………………………………………………………………………………Page 17
9. Actual Key Risks………………………………………………………………………………………………………….Page 18
10. About Team…………...…………………………………………………………………………………………………..Page 19
11. Checklist………………...…………………………………………………………………………………………………..Page 19

### Executive Summary

Product Name: Hardware Nanny

Our proposed webstore project aims at providing a straightforward, accessible and eye pleasing interface combined with an efficient management system for the inventory and services of a webstore application. The objective of this project is to resolve the issues that may arise when trying to manage a large inventory database by providing a software solution for keeping track of part or job requests and management of part information and quantities. This reduction in manual input and management leads to more time for other tasks such as accurately monitoring the efficiency of activities and services that take place with the university’s hardware tools and components.

This project is designed with its target audience in mind which would allow it to be implemented and adjusted to meet the requirements of any university seeking to manage hardware parts and equipment use. The general idea is that although the system keeps track of everything within the database, given the appropriate credentials, the admins or faculty will be able to access and manipulate the data and items within the inventory while students would have access to specific features. The simple interface of the system has functions such as adding a part, creating a kit, and removing or updating an item. It also extends to functions such as adding or removing the vendor for a specific part, checking threshold levels, and processing orders or special jobs. Furthermore, the system will collect previous and current data, use algorithms to estimate parts needed and compute average cost. Another functionality would be to enable user feedback to better address issues such as which parts are the best quality and which are the most cost-effective. The system would keep track of the minimum parts needed within the inventory. All of the systems various functional and nonfunctional features and high-level architecture that define the system along with its unique aspects makes this project a useful and pragmatic tool for its various stakeholders.

On the whole, this system offers an effective solution to better manage the databases and allows users to focus on tasks that computers cannot replace. It is a secure system that is easily accessible on or off campus. The interface is intuitive to navigate and without complex functionality that would cause trouble for new users trying to grasp the systems’ operation. The totality of the aspects that make this system viable are detailed below where we explore what makes this system serve the user’s needs. All in all, Hardware Nanny strives to be an invaluable software system that streamlines and improves the hardware management process.

### Competitive Analysis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Companies** | Hardware Nanny  (our product) | Newark.com | Amazon.com | Homedepot.com |
| **Safety** | Needs Florida Atlantic credentials to use | Is Norton Secured | Secured by Amazon | Secured by The Home Depot |
| **Development Cost** | Free | Upwards to 5 thousand dollars | Anywhere from $10,000-$30,000 | Anywhere from $10,000-$20,000 |
| **Applicability** | Florida Atlantic Engineering Students | Customers needing hardware parts | Literally anything, especially fast and efficient | People looking for home improvement solutions |
| **Difficulty of Use** | Very straight forward and organized | Quite easy, it is very well  labeled | Easy. Can browse categories or search for items | Easy. Can browse categories or search for items |
| **Interface** | Very basic, with little clutter and very straight forward | Clean boxes with different sections of hardware types | Very lively page with recommended products | Very minimalistic with their sales displayed largely |
| **Specialty Jobs** | 3D printing, laser cutting, PCB | Allows custom orders and kits | Allows to sell your own products | Will install certain products in your home |
| **Rent parts** | Ability to rent specialty tools/parts | No rentals | You can rent books | Large amount of equipment to rent |
| **Item request** | Customers will be able to request a new item be ordered | Special orders for most to all hardware parts is available | No item requests | Requests for certain items is available |
| **Build Kit** | Admins create kits for customers to purchase together | Kits of hardware from different company available | They can bundle certain products to be shipped together | They sell a kit of products to complete a room in a house |
| **Special Features\*** | Product reviews.  Engineering news.  CEECS Calendar. | Buy large quantities of hardware and very specific products. | Very good algorithm for recommending products | See products in model homes before purchase |

**NOTEWORTHY FEATURE SUMMARY**

Our Product Hardware Nanny provides a very simplistic way to order parts as an engineering student at Florida Atlantic University. It was developed to be easy to use with an organized platform for students to use. It can only be used by those who have proper Florida Atlantic credentials in order to keep our webstore safe and not abused. The user Interface of Hardware Nanny is simple to use with little clutter from unnecessary pop ups throughout the site. It is a very straightforward but informative interface. Our webstore offers specialty jobs such a 3D printing, laser printing and printed circuit boards. These jobs are bound to set us apart from the competition and make promise towards great customer approval. Along with these specialty jobs our webstore will offer rentals like not all but some other competitors. These would include parts that are scarce or parts that are too expensive to buy outright for short time use. If a item in the webstore is unavailable at that time they go look for it, there will be a feature that allows the customer to request this item for it to be available at the in person store at FAU. The administrator of the webstore will be able to compile custom kits that will be available for certain classes and events going on at FAU. This process of kit creation will be easy and very helpful for students that take part in these events or classes. The parts that will be available from the webstore will also be paired with reviews from students that have used them. The sight will also feature the earlier mentioned events on a calendar going on in the CEECS department with links to possible kits need for these events.

### Data Definition

**Hardware Parts**

Are the central stock keeping units that can be procured without any monetary compensation and have descriptions and specifications stored within the database of the software system where information including the short and long description, quantity on hand, serial number, prices and vendor information can be found each Hardware part is distinguished by admin or faculty inputted data and can be removed and added to the software system by authorized personnel.

**Part Request**

Is an entity that reflects the need of the user to get a part from the inventory. The parts request must be completed through the Hardware Nanny software all needed details must inputted to trigger the part request event. The part request event provides a status viewable to the user to check in the user account. This state of the Part request can be one use time limitation only or can fulfilled or denied. The part Request is administered by the admin on whether to respond by ordering the part or notifying the user that the request cannot be fulfilled by returning the part request status of unavailable.

**Specialty Job Request**

Reflects the need of the student to initiate a relevant job such as Laser cutting 3D-Printing, Printed circuit Board. This request can be placed by the user with proper formatted necessary documentation files and other relevant comments through the Hardware Nanny software system. The Specialty Job Request provides an estimated time for the job to be completed. Student users must have all the necessary schematics and documents specified during the Specialty Job Request application with proper file types needed to successfully initiate a job to be done within an admin determined grace period. The status of the Specialty Job Request is viewable through the a user account.

**Rental Request**

Is the software means to allow students and or permissible users to rent a part from the parts inventory list for a certain duration. After a Rental Request has been initiated the user account information reflects this rental request state and the agreement terms. The terms of the Rental request are controlled and enforced by the admin. Only selected inventory will applicable for rental requests to users.

**Part Review**

A place for staff and students to comment on part issues and a way to analyze if the part contributed an appropriate solution to the user’s institutional or research needs. This star rating system allows for a 550-character description of the users concerns or satisfaction with part used. The Part Review is managed by admin personnel.

**Hardware Nany Client**

Is available to students and staff based on permission given at the university status. Perry and selected staff administer this client and monitor activates and update information and provide authorizations and responses with ease. The client is available on permissible computer terminals in the Engineering Labs and in other designated areas determined by the admin and the university. The client is designed to provide a solution to hardware inventory management access and use as well as allow special job services.

**Status**:

Is the present state of a part, rental or specialty job request which can be (out of stock, Available, requires admin permission, Confirmed, Single Use only, Unavailable) this indicator triggers various events such as allowing a user to request, rent or add a part and or kit or preventing them from doing so. These status states result in other procedures such as confirming a rental with specified terms or recommending the user to contact admin, rejecting the request or otherwise the requested or ordered part/kit is ordered. The status change for a particular item or items will be reflected database details such as quantity on hand will is updated accordingly. The status of a special job request is also managed by admin and viewable to the admin the user that placed the request.

**Quantity on Hand**

Is the number of a particular part-based admin manual entry, automated new order update entry when part is received and in stock or depleted and needs to be reordered. The quantity on hand is the basis to compare with the frequency of when quantity on hand is stagnate or is regularly depleted. The part/kit quantity on hand data serves as a way to inform admin how often the item is used or lack thereof.

**Short description**

This is a description in the database viewable to the students, modifiable and viewable to the faculty and admin provides a brief description of the part, kit for which the user is requesting, or renting or creating, for additional information beyond the short description can be found in the long description.

**Long description**

This is a description expands upon the short description product details and is in the database viewable to the students, modifiable and viewable to the faculty and admin provides a detailed description of the part, kit for which the user is requesting, or renting or creating.

**HN Kits:**

These special combinations of hardware parts built by faculty or admin these kits are built at the staff’s discretion and can be modified as needed with different part combinations. The customized kits purposes as an efficient way to package together applicable hardware components from the primary inventory and to streamline applicable part kits for distribution for instructional use

**Location in lab**

A location such as the cabinet, drawer, area inside drawer is the specific position description given for specific the parts and/or kits that faculty and admin can retrieve as needed. This position information will enable better organization and accessibility of inventory parts and kits. This location in the lab is viewable to a permitted user after logging into their respective account.

**Cost averaging**

This is special operation within the software system that involves a condition when the cost of a product changes as a result of restock the system takes into account if there are products left over from the last order and reflects the new cost average.

**Single Use Only**

A special status that prevents repeated ordering and gives user the one time opportunity to request a part or kit. This status is given only to certain inventory items which is determined by the

admin

**Parts Vendor**

This entity contains the Address, email and relevant contact information describing the vendor as well as an or Account # with the vendor, Parts offered by vendor, Product/s and pricing. A vendor is created, removed and updated by the faculty or admin. The vendors part options are comparable with other vendors listed when parts searches are initiated by the user. A student account does not have access to this information.

**User Account**

Upon using the Hardware Nanny client, a user must create and access a user account to allow them to utilize the features the Hardware Nanny client provides. This user account is structured in three-tiers that allows students, faculty, and admin with certain permission allowances depending on account level status. The account setup requires qualifying credentials such username and password as well as relevant user contact information. User accounts are viewable and manageable to admin. The accounts created by the user serve as a basis to monitor Hardware Nanny client activity history.

### Overview, Scenarios and use cases

Customer use cases:

1. **Create an account (first visit) and sign in**
   1. A simple login page will be used to allow the user to enter the web store or they can they can be automatically logged into the system through the University website
   2. Customer uses Email and Password to sign in, this unique login feature will be established through FAU single sign using the FAU login credentials in in the future
2. **Order parts**
   1. Customer goes to website and signs in.
   2. On the profile page or other user authorized pages( excluding the form and confirmation pages) the customer clicks on the ‘Inventory’ link on the top menu bar
   3. Customer searches for the part they want (or browses the entire list)
   4. Customer adds desired part to their cart and indicates the part quantity needed and reviews the cart and may checkout, remove the item or continue shopping.
   5. Customer submits the order and views the confirmation page with their confirmation number and can print this page if needed
   6. Within 24-48 hours if approved an email is sent giving the pickup location is provided else an email is sent that details an order rejection
3. **Request new item**
   1. Customer goes to website and signs in.
   2. On the profile page or other user authorized pages( excluding the form and confirmation pages) the customer clicks on the ‘Inventory’ link on the top menu bar
   3. Customer searches for an item and does not find the item.
   4. Customer clicks on “Request new Item” from and fills in all the necessary fields for the item
   5. Admin approves or rejects the request
4. **Request a job**
   1. Customer goes to website and signs in.
   2. On the profile page or other user authorized pages( excluding the form and confirmation pages) the customer clicks on the ‘Job Request’ link on the top menu bar
   3. They are then brought to the “job request form” page and fill in all the necessary fields for the job
      1. 3D Printing
      2. Printed Circuit Board (PCB)
      3. Laser Cutting
   4. Customer uploads all the proper files and a comment about the job and the staff will complete it for them.
   5. Within 24-48 hours if approved an email is sent giving the pickup location is provided else an email is sent that details an order rejection
5. **Rent an item**
   1. Customer goes to website and signs in.
   2. On the profile page customer clicks on the ‘Inventory’ link on the top menu bar
   3. Customer searches for the part they want (or browses the entire list) and find the item is only available to rent for a certain duration.
   4. Customer clicks “rent” button and is brought to the rental agreement page.
   5. Customer submits the order and then picks up the item from the lab and then returns the item after the rental time has expired.

Staff/ Admin use cases:

1. **Everything a customer can do**
2. **View orders/ Jobs/ Rentals** 
   1. Staff member signs in.
   2. On the profile page or other user authorized pages(excluding the form and confirmation pages) the customer clicks on the “Orders” link on the top menu bar
   3. Staff member clicks on ‘order details’ for the first ‘open’ order, Job, Rental gets each part for the order ready or job prepared if the order is approved or rejects the order with a comment.
   4. Upon approval customer can picked the items up or rental or wait for the job request to be done. Staff member/ Admin marks the order as completed. If rejected then the customer receives an email about the rejection. The decision made on the order is reflected in the view orders staff page
3. **Create kits**
   1. Staff member/ Admin signs in
   2. The customer creates a brand-new kit with “Add new kit” button on the left side of the browse inventory staff/Admin page and fill out the ‘Create Kit form’ with proper information in the various fields
   3. The customer can add to an existing kit or add items to a kit
      1. On the profile page or other user authorized pages (excluding the form and confirmation pages) the customer clicks on the “Inventory” link on the top menu bar
      2. Customer searches for the part they want (or browses the entire list)
      3. Customer adds desired part to their cart for the desired kit and indicates the part quantity needed and reviews the cart and may checkout, remove the item or continue shopping.
      4. Customer submits the order and views the confirmation page with their confirmation number and can print this page if needed
   4. The kits created including details on individual kits can be reviewed on the Kit Derails page
4. **Add/edit/remove inventory**
   1. Staff member/Admin signs
   2. On the profile page or other user authorized pages (excluding the form and confirmation pages) the customer clicks on the “Inventory” link on the top menu bar
   3. The click the ‘Add New Item’ button on the Browse Inventory page to create a new item in the inventory
   4. They can modify or remove an existing item by editing it in the Item details page
      1. Staff member finds desired part and clicks “edit part”, edits the part information, and clicks “save”
      2. OR staff member finds desired part and clicks “remove part”
      3. OR staff member fills in all relevant fields and clicks “add” (or uploads spreadsheet of multiple items)
5. **Add/edit/remove vendor**
   1. Staff member or admin signs in
   2. On the profile page or other user authorized pages (excluding the form and confirmation pages) the customer clicks on the “Vendor” link on the top menu bar
   3. Staff member finds desired vendor and clicks “edit vendor”, edits the vendor information, and clicks “save”
   4. OR staff member finds desired vendor and clicks “X” to remove the vendor
   5. OR staff member or admin clicks the ‘Add new vendor’ button on the Browse vendors page and fills in all relevant fields and clicks “Create Vendor”

Admin use cases:

1. **Everything staff/customer can do**
2. **Create/edit staff or customer accounts**
   1. Admin signs in
   2. On the profile page or other user authorized pages (excluding the form and confirmation pages) the customer clicks on the “Profiles” link on the top menu bar
   3. Admin finds desired account and clicks “add/edit/remove”
   4. Admin enters/edits account information and clicks “save”
3. **Approve new item requests**
   1. Admin signs in and clicks “orders”
   2. Admin clicks “approve” and orders new item.
   3. OR Admin clicks “Reject”

### List of high level functional requirements (with Rating System 1,2,3)

1. **Accounts**
   1. Users must be able to create an account via FAU single sign in and login FAU credentials using email and password **1**
   2. Users will have one of three levels of access (customer, staff, admin) **1**
   3. Users must be able edit the information in their account, which includes: **2**
      1. Name, z-number, email, phone# (optional), college, department, graduation date, classes taken (3 letters, 4 numbers), password **1**
   4. Admins must be able to edit Staff and User accounts **1**
2. **Part list**
   1. Users must be able to see the entire list of parts or search for a particular part
   2. Parts will include: **1**
      1. Part #, short description, long description, multiple images, multiple data sheets, location in lab (cabinet, drawer, area inside drawer), quantity in stock, multiple fields for price, part # and price for each vendor, cost averaging, rental item only, one time use only. **2**
   3. Staff and Admins must be able to add/edit/remove inventory **1**
   4. Staff must be able to add parts into a kit or create a new kit **1**
   5. Users must be able to request new parts **3**
3. **Cart**
   1. Users must be able to add items to a cart, which can then be ordered. **1**
   2. Cart will include the number of each part ordered, the quantity can be changed within reason unless items is single use. **1**
   3. Users must be able to edit their cart such as remove or add items/jobs from the cart or continue shopping. **1**
4. **Orders**
   1. Customers must be able to view their own completed and in progress orders **2**
   2. Staff members must be able to view all completed, rejected and open orders. **2**
   3. Staff members must be able to mark orders as complete and provide pick up location details or reject them with a comment **1**
5. **Vendors**
   1. Staff and admins must be able to browse the entire list of vendors or search for a particular vendor **1**
   2. Vendors will include: **1**
      1. Website/other contact information, account #, parts offered by vendor (product # and price) **2**
   3. Staff and admin must be able to add/edit/remove vendors **1**
6. **Jobs**
   1. Users must be able to request specialty jobs to be completed by the staff and upload the necessary files and job information derails using the Job Request Form. **1**
   2. Staff or admin must be able to view job requests the browse orders page, and as needed download the necessary files, and mark the job as completed or reject the request with a comment. **1**

### List of high level non-functional requirements

**Usability**

* 95% of all users will be satisfied with the usability of the product.
* The system must be easy to use by both Perry, Students, Faculty and have minimal to no issues when used and is operable for a majority of users without a need for application support documentation.
* The system must be quickly accessible to Perry Students Faculty & other authorized personnel on authorized computer terminal workspaces.
* The system must be intuitive and simple in the way it displays all relevant data and relationships.
* The menus of the system must be easily navigable by the users with buttons that are easy to understand.
* The Hardware Nanny System should operate using all faculty, students and admin operating systems and function and install properly.
* Students, staff and admin data is taken for all services used to monitor frequency enabling billing for outside departments and overall monitor services used over time.
* Recommender systems or recommendation systems are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that a user would give to an item.

**Reliability**

* The System must give accurate inventory status to the user continuously. Any inaccuracies are taken care by the regular confirming of the actual levels with the levels displayed in the system.
* The System must successfully add any, vendors or special occasions given by the user and provide estimations and inventory status in relevance with the newly updated entities.
* The system must provide a password enabled login to the user to avoid any foreign entity changing the data in the system.
* The system will provide the user updates on completion of requested processes and if the requested processes fail, it should provide the user the reason for the failure.
* The system should not update the data in any database for any failed processes.
* In the event of the user cancelling or quitting the process “updating part information” any changes made by the user will be reversed.
* The system is required to support multiple terminals simultaneously. The system should handle reasonable number of users without break or inconsistency

**Performance**

* The system must not lag, because the students, faculty and the admin using it don’t have down-time to wait for it to complete an action.
* The system must complete updating the databases, when a user creates an account, when students orders or rents parts, when staff adds items to inventory or adds a new vendor when admin create/edit staff accounts, create/edit customer accounts successfully every time the user requests such a process.
* All the functions of the system must be available to the user every time the system is turned on.
* The calculations performed by the system must comply according to the norms set by the user and should not vary unless explicitly changed by the user.

**Supportability**

* The software is designed such that it works even on systems having the minimum configuration.
* The system is adaptable even if additional plugins or modules are added at a later point.
* The data can be exported to the admin so as to make the system more portable.
* The system should be operational on all designated University computers terminals and accessible off campus with specific off campus restrictions including a different ip address that distinguishes the user.

**Security**

* Login requirements: an account must be established with university email and credentials and verified during the account creation process.
* Password changes and forgotten password recovery are permissible to all users
* For security purposes all users must sign in with email and password after 4 attempts the system locks the user out and has the account owner reset their password and when doing so has them answer a security question.
* Password requirements: passwords must be at least 8 characters in length, cannot be based on dictionary words/common names, and must contain at least 3 of the following 4 types of characters: lowercase letters (i.e. a-z), uppercase letters (i.e. A-Z), numbers (i.e. 0-9), special characters (e.g. -=[]\;,./~!@#$%^&\*()\_+{}|:<>?)
* Inactivity timeouts: a user of the system is automatically logged out of the software after a period of 15 minutes of inactivity and needs to log back in as they are notified of the software timeout.
* In order to distinguish on campus and off campus users 2 different designs will be used to differentiate users with ip addresses in campus and out of campus. Restricting certain functionality based the type of ip address design used when software system is accessed.
* DoS attack protection strategies will be employed such as DoS attack identification and detection techniques to differentiate between legitimate and malicious traffic. Using activity profiling procedures and mechanisms which measure average traffic rates and flags significant increases in traffic to identify when an attack is underway. The use of throttling and rate-limiting technologies will be used to reduce the effects of a DoS attack including such tactics as a response mode that stops all new inbound connections in the event of a DoS attack, allowing established connections and new outbound connections to continue.

**Storage**

* Throughput storage of the web store application should have approximately 5 megabytes.
* The bandwidth to be determined must meet approximately 100Mbps per 1,000 users or 100Kbps and will require 1Gbps per 1,000 students or 1 Mbps per user
* The system should be able handle at least the total student population that have accesses to the system making requests special jobs and products per hour.
* A dedicated cloud server is used to store data with at least 8 GBs of storage space.

**Fault tolerance**

* Recovery process can be initiated by the admin personnel this process is prompted to executed during instances when the security has of the software has been compromised
* Recovery time depends on the database size growth potential approximately a 1 hour to 2 hours to completely analyze the system for failures or security compromises
* Backup frequencies: occur every week, or at the end of every day during peak times with new shipments or large request orders are placed.

**Constraints**

* The Hardware Nanny Software Systems is most influenced by the admin and faculty that regulates various system functions at their discretion and Its database is moderated by both admin and faculty the admin in particular manages all customer accounts and requests and overall inventory.
* Student users are given limited access and retain the most restriction access when using the specialized inventory management software.
* Monetary transactions are not made through the systems only data about the records of requests are collected to reflect request and service use history.

**Implementation**

* The System User Interface is built on Adobe Brackets version 1.12 using Bootstrap and JavaScript.
* The Programing is done in on Adobe Brackets version 1.12.
* The Database is implemented on the MySQL

### High level system architecture

Student

Staff

Manager

User Login Interface

Parts Inventory

Shopping Cart

New Item Request

Inventory Manager Interface

User Login Interface

Account Manager Interface

### 

### 

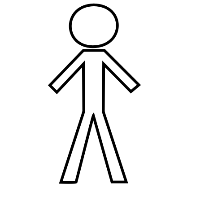
### 

### 

### 

**UML Class Object Diagram Inventory/Orders**

UML (M3 Add Item to Inventory Feature)



### Customer -Admin

### Database Organization

* Database Organization/Schema:

We will utilize at least 5 databases:

* + Logins – allows user login/sign in, used in M3
  + Profiles – Contains user profiles of webstore
  + Orders - Contains user orders placed through the webstore
  + Inventory - Contains inventory items of webstore, used in M3
  + Vendors - Contains vendor information of webstore
* Media storage Plan:
* Images such as jpg, png, and other image files types will be stored in the inventory database
* Additionally, we will also store other relevant files that pertain to job request submissions through the orders database in the beta release
* Search/filter architecture and implementation:
* Inventory Search capability
* By part name
* By part number
* By price
* By kit name
* Vendor Search capability
* By vendor name
* By item
* Profiles Search capability
* By account name
* By account number
* By class of account holder
* Orders Search capability
* By person the place the order
* By order number
* By item ordered
* Describe and define at high level any major APIs that you will Create
* We will create our own relevant APIs via bootstrap 4
* All current APIs are in the beginning stages and in development at this time
* Describe any significant non-trivial algorithm or process:
* Rating: star system for each item/kit
* Ranking: most popular item searched or searched by similar account holder type
* Automatic prioritizing of items: by which was most recently viewed or viewed most frequently

### Actual Key Risks

* Skills risks
* Various groups members have a range of skills with differing strengths & weaknesses and must be balance during the development process
* Many of us have not had formal extensive experience in backend development and must compensate for that knowledge absence
* Many of the group members have varying degrees of front end experience
* Schedule risks
* A major challenge is scheduling time for the project in compliance with all members varying availability
* Distrusting tasks swiftly to achieve project submission goals is a difficult task even with internal deadlines
* Organizing and delegating who can best complete what in an efficient appropriate manner is difficult
* Technical risks
  + Dealing with the 3-Tier sign in/ FAU single sign in
  + Operating with the appropriate upmost security approach for University standards
* Teamwork risks
* Many of us must do tradeoffs and balance our workload
* Desired participation VS ability or skill limitations
* Legal/content risks
* Not applicable/ to be determined

### About team

* Alicia Mitchell – Scrum Master
* Sergio Ramirez – Product Owner
* Bradley Rubenstein – Back End Lead
* Guimel Gonzalez Hernandez – Front End Lead
* Ryan Juraidini – GitHub Master

### Check List

1. Team Communication method: DONE (WhatsApp Group Chat)
2. Outside of the class team meeting: DONE (Tuesdays & Thursday 3-5pm)
3. Front end and back end leads: DONE (Guimel, Brad)
4. Github Master: DONE (Ryan)
5. Team selected backend framework: ON TRACK
6. Skills of each team member defined and known to all: DONE
7. Team lead ensured that all team members read the final M1 and agree/understand it before submission: DONE