Department of Computer Science & Engineering
The University of Texas at Arlington

Project Charter
CSE 4316: Senior Design I
Fall 2021

Team GI
Inventory Database

Tochy
Ian
Jonathan
Jake Ryan
David Rademacher

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	9.28.2021	Tochy	document creation
0.2	9.28.2021	DR	sec 3, 12, 13
0.2	12.04.2021	Jonathan Banda	section 1, 2, 3, 4
0.1	12.04.2021	Tochy	section 5, 8, 13
0.3	04.18.2022	Jonathan Banda	section 4, 9, 11, 14

Contents

List of Figures

1 Problem Statement

The senior design labs is in current need of managing the items they store, lend out, and recieve by students enrolled in senior design. Senior design lab would like to be able to easily manage and review all of the equipment that current students borrow. Easy reporting and accurate information about students is critical in addressing items not returned.

2 Methodology

We will build a database backed website to host as a centralized source for the sennior design lab to gather and store information regarding items lent out. The system will need to work in a cyclic mannner allowing a user to identify the student checking in/out, proccess item to be checked in/out, and update the system according. Other features such as report generation and reminders may be implemented to assist the senior design lab in recovering borrowed items from students or take action accordingly.

3 Value Proposition

Managing inventory is one of the most crucial administrative tasks of a well-organized institution. Under the constraints of a budget and with thousands of users, such institutions must optimize their tools and processes. The implementation of a database interface that handles inventory management will make the process more efficient (requiring fewer man-hours to complete) and less prone to error. Database systems are optimized for the kinds of tasks most needed by the senior design lab: creation of data,removal of data, and sorting and presentation of data. When combined with an accessible interface, the General Inventory tool provides a streamlined solution to inventory management that will save costs and allow the program to allocate resources where they matter the most.

4 Development Milestones

- Project Charter first draft - October 4th, 2021
- System Requirements Specification - October 25th, 2021
- Architectural Design Specification - November 15th, 2021
- Formalization of Senior Design Lab requirements - December 6th 2021
- Detailed Design Specification - Febuary 15th, 2022
- Demonstration of database functionality - Febuary 2022
- Prototype scanning system - Febuary 2022
- CoE Innovation Day poster presentation - March 2022
- Demonstration of Full basic application running - April 2022
- Final Project Demonstration - April 28th, 2022

5 Background

The UTA CSE program currently does not have a method to track the items stored within the lab. The CSE department is in possition of various types of expensive equipment such as computer, virtual reality equipment, and many other items. These items are lent out to student within the senior design classes to aid them in their project. Without proper knowledge of what item is currently being used could cause scheduling collisions, which may as well delay a groups progress. The CSE department also takes a great risk lending these items out to student. Without knowlegde of which student is currently using the item means if the item is lost, the blame would fall on mulitple people. Being able to know who has what specific item checked out would help bring an oversight to these particular problem. Being able to know how many equipment is avaiable for use, and who is specifically using the ones lent out would be a great improvement to the UTA CSE depatment. Since students are already given student ID, we can use the information stored in them to help simplify the database management. This also means that students who do not have their ID on hand will not be able to checkout any equipment from the CSE lab.

6 Related Work

The current state of the art with respect to our product includes services such as Amazon Web Services (AWS) [1], FishBowl [5], and tableau [7]. The technologies used to develop similar products with respect to ours include front-end and back-end such as React [6] and django [4] through REST api. FishBowl [5] provides an interface that can manage inventory, "Track inventory with lot/serial numbers, expiration dates, and more to efficiently manage multiple warehouses" [5] which correlates to the inventory management aspect of our product. Tableau [7] is a data visualization software tool that helps users create engaging visuals, "Tableau helps people see and understand data." [7].

There is currently solutions available in the form of commercially available products and academic research. In regards to commercially available products, there is a plethora of task specific software that can meet out sponsors needs. That being said, it would take multiple software products from different vendors in order to fulfil the needs of the sponsor. A solution for storing users data or data in general can be solved with the use of a cloud database. Amazon [1] offers a relational database service that can be used to store data on the cloud, "easy to set up, operate, and scale a relational database in the cloud." [2]. In regards to academic research there are models available that can detect text from an image. Optical Character Recognition (OCR) is the appropriate term for image to text as described, "...technology that enables you to convert different types of documents, such as scanned paper documents, PDF files, or images captured by a digital camera into editable and searchable data." [3]. This technique solves the problem our sponsor has over converting physical document data into a database.

Although commercial software products are readily available in today's markets, they fail to meet every need of our sponsor. Our sponsor wants a system that can store user information, extract information in printable form, process physical documents and store documents. They also want a system that has a hierarchical structure in terms of access to certain features within the system. There is not an existing solution that bundles all the requirements of the sponsor. The existing solutions would cost too much per license and the sponsor would not use many of the features offered within the software product.

7 System Overview

The entirety of our system will rely on calls made to the database. Therefore in order to build such system we need the foundation of our architecture to be comprised of a database layer. The database layer plays an important role in maintaining integrity and functionality of the entire system. Working along side the database layer should be a login layer. The login layer serves its purpose in encapsulating/restricting privileges granted to users. This layers is essential in integration of the system. Other than those two layer there is only one other high level layer that must be taken into consideration, which is a dashboard layer. The dashboard layer consists of all the sub modules features of the system. This layer handles the flow and functionality of the systems features. The Layer that will put everything together is the Web Application Layer which in other words is the "api" layer that handles request to and from the database that are made on the web app. It will use the REST framework in order integrate properly.

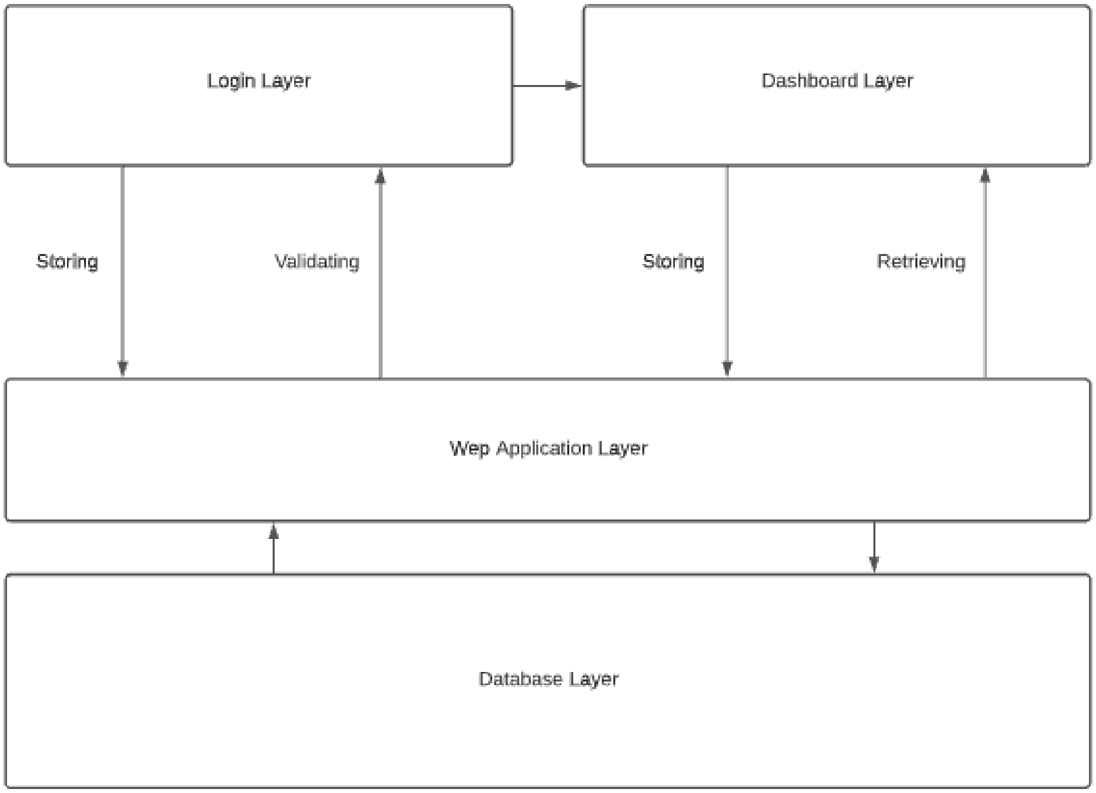


Figure 1: Example Functional Overview

8 Roles & Responsibilities

The stakeholders include CSE department, the instructor, and all those individuals working on the project. CSE department is concerned with the software product outcome making sure it satisfied all of their requirements. They will be positively impacted by the development of the General Inventory system. The instructor is concerned with the quality of work performed during the project planning and development according to ABET standards. The team will be impacted as a result of their work. The team working on the project is concerned with delivering a working software product to the sponsor (CSE department) while satisfying requirements. The team will be receive feedback from the instructor throughout the development of the database system.

The point of contact between our team (General Inventory) and CSE department will be everybody. Everyone will convey thoughts and concerns our sponsor may have during the planning and development of the General Inventory system. Regardless in the case that Tochy cannot convey the teams' question and concerns someone else may assume that role. In respect can the sponsor decide on someone else being the point of contact if they feel it is necessary in order to accelerate the development of the software product.

Everyone working on the General Inventory system include Jonathan, Tochy, Ian, David, and Jacob. At the moment of this writing, the roles and responsibilities have not be assigned. That being said we are currently breaking up all work related to the project equally with random assignment per sections. In the future we want to break up the project into two teams; front-end and back-end. The individuals working on the front-end aspect of the project will be responsible in making the web application user friendly and dynamic. The individuals working on the back-end aspect of the project will handle data manipulation, data integration, and data migration.

In regards to product owner, everyone in the team will have equal opportunity to be the product owner. Everyone contributed to defining the product backlog therefore everyone has input and responsibility to ensure that the teams meets the goals. In regards to scrum master, we do not have a scrum master and are going to support each other through the planning and development phases of the project. Everyone is responsible equally to make sure the team members are following agile practices. Therefore, both product owner and scrum master role will not necessarily change as the roles will be filled by each member of the team through the completion and delivery of the software product.

9 Cost Proposal

The starting budget for this project is the departmental funding given to us, which is currently considered to be more than enough. This amount is $ 800 which with the team can secure usage of, or creation if required, a website and database system. We will need to allocate funds for software needs and hardware needs in order to support the development and deployment of our system.

Software to implement or use prebuilt systems has been set aside, however our sponsers security concerns may require us to implement direct hardware control and may cause the budget to potentially need expansion or go closer to our current maximum allotment. Hardware to be used in testing functionality of our systems image to text feature as well as inventory management flow.

9.1 Preliminary Budget

Catagorized potential item list of expenses needed for project to reach completion.

Items and Catagories	**Amount($)**
Hardware	
—-Database	0
—- Website/Front-end	0
—- Printer	130
—- Printer Paper	20
—- Barcode Scanner	50
—- Magstrip Swiper	20
Total	220

Table 1: Overview of expected or potential costs

9.2 Current & Pending Support

Currently we are only being supported by departmental funding which gave us a budget of $800. At the moment we have not reached out to our sponsor regarding additional funding. We do not plan on going over the budget at the moment. However if circumstances arise and we need additional funding we will request more funds from our sponsor.

Revenue Source	**Amount ($)**
Department Funding	800
Sponser Funding	TBD
Total	800

Table 2: Overview of Funding

10 Facilities & Equipment

The facilities and equipment that a project like this one requires is quite minimal, as the only thing that will be required for the team to procure will be hosts, one for the database and another for the method of with which we access said database. The users who will be using this are expected to have devices that will be able to interface from. The database, and potential other hosting measures can be implemented in a variety of ways with which can require equipment to be run, or to be outsourced. Software licenses are not expected to be required, however in researching the specifics on implementation of the system it may be beneficial to do so for one reason or another.

Specifically, the database system in of itself will either need to be setup with a machine which will need maintenance or need to be outsourced from a service that would provide one like firebase database as an example. The way of accessing the database is more complicated, as the implementation can theoretically be done directly however in practice outsourcing or having a machine run a website or some other intermediary is planed for from with which to connect to it. This is most likely a must due to requirements, doing so does simplify the implementation however potentially given the broadness of the requirements from our sponsor doing without is open to implementation if budget requires it.

Testing grounds, makerspaces, and lab spaces are not needed for this project due to its nature. Perhaps a place to store and operate the computers from with which the database/access-point will be needed but the potential to outsource and not needed is also an option that bears consideration and might be the ideal decision to go with. Security might require us to choose to host with someone or create machines to host, this decision will depend on specifics of sponsor requirements and implementation choice.

11 Assumptions

The following list contains critical assumptions related to the implementation and testing of the project.

- The sponsor will provide a list of what information will need to be stored per user.
- The sponsor will provide a list of items that will be counted as inventory.
- The sponsor will provide a requirement list stating what items should be named in the system or agreed on the team's naming conventions.

- The sponsor should be able to connect to the UTA WiFi system to use the web application online.
- Inventory will be available to perform label print and scanner test into the database.
- Scanner and label printer should arrive once before the database schema is complete.
- Hosting services will have been purchases before testing phase starts.
- Web page naming should be provided by the sponsor prior to purchasing web address.
- The users should understand how to use the web app before delivery of software product.
- Login credentials (master user) should be defined before starting code sprint 1.
- We are assuming that the database host is secure enough.
- Assuming sponsor will be using modern operating system to run web application.
- The software product will be done early enough to test product with customer.
- We are assuming someone will take over maintenance after completion.
- We are assuming that the printer can connect to the uta wifi

12 Constraints

Being focused only on software solutions, the General Inventory project faces somewhat fewer constraints than if hardware systems were required to be designed. Given that the project will be entirely composed of code, there are still several bounding forces by which the team will abide over the course of development.

The following list contains key constraints related to the implementation and testing of the project.

- Data stored may fall into various classification levels, and security clearance laws/requirements must be adhered to
- Final prototype demonstration must be completed by May 2021
- Total development costs must not exceed $800

13 Risks

The risks to the General Inventory software are generally related to the standards of the CSE department. Due to the nature of work, the project should be treated like a government contract to avoid security clearance conflicts.

The following high-level risk census contains identified project risks with the highest exposure. Mitigation techniques are discussed in table underneath highest exposure project risks.

Risk description	Probability	Loss (days)	Exposure (days)
Security clearance issue	0.50	14	7
Delays in communication about software specifications	0.25	8	2
Team mate gets covid	0.1	28	2.8
Team mate drops course	0.05	60	3
Has to restart due to customer changing requirements	0.1	20	2
Team mates learning new langugages or frameworks	0.9	15	13.5

Table 3: Overview of highest exposure project risks

Risk description	Mitigation Technique
Security clearance issue	Discuss with customer early on to clarify any clearences needed
Delays in communication about software specifications	keep many communication lines open and discuss with customer frequently
Team mate gets covid	Do peer programming or code reviews so that everyone has a general idea of what all code is doing
Team mate drops course	Do peer programming or code reviews so that everyone has a general idea of what all code is doing
Has to restart due to customer changing requirements	Clarify and formalize customer requirements and not budge if they ask for new features
Team mates learning new langugages or frameworks	Put youtube video tutorials at double speed

Table 4: Overview of mitigation techniques

14 Documentation & Reporting

14.1 Major Documentation Deliverables

14.1.1 Project Charter

Update Cycle: The Project Charter will be updated every 2 sprints, or in the event of drastic changing requirements. Before a new project charter version is put out it will require overview and approval from all team members regardless of if they were part of putting the new changes or not. The Project Charter will also be updated and reviewed in the light of negative feedback from the professor or customer on our current process.
Initial Version completion date: 10/4/2021
Final Version completion date: 4/26/2022

14.1.2 System Requirements Specification

Update Cycle: The System Requirements Specification will be updated in the case that the requirements change. The requirement change could either be an internal choice by the team, for example changing the database system we are using, or it could be external, for example the customer changing what they want later into the project.
Initial Version completion date: 10/25/2021
Final Version completion date: 4/26/2022

14.1.3 Architectural Design Specification

Update Cycle: The Architectural Design Specification will be reviewed every sprint to make sure we are following the framework, and updated should it be needed. The goal is to not have to update the Architectural Design Specification due to the fundamental nature of the document. A change in architecture would possibly mean a total rewrite of the code.
Initial Version completion date: 11/15/2021
Final Version completion date: 4/26/2022

14.1.4 Detailed Design Specification

Update Cycle: The Detailed Design Specification will be updated every sprint to accurately depict our design choices. This document will be the most frequently updated due to design choices and brain-

storming and prototyping.
Initial Version completion date: 2/15/2022
Final Version completion date: 4/26/2022

14.2 Recurring Sprint Items

14.2.1 Product Backlog

The items will be added to our product backlog on Jira, Prioritized by dependencies and essential features, decided by the product owner (Tochy) and with veto power in the group. Jira will be used to maintain the product backlog and share with the teams. Should Jira not work or become impractical we will switch to another software.

14.2.2 Sprint Planning

Sprints will be planned by the team, so that each team member can give their availability for the sprint so we can decide how much work to designate to each member. Their will be 8 sprints in total.

14.2.3 Sprint Goal

Our Product Owner(Tochy) will decide the sprint goal and if he decides the customer needs to have input he will contact them as necessary.

14.2.4 Sprint Backlog

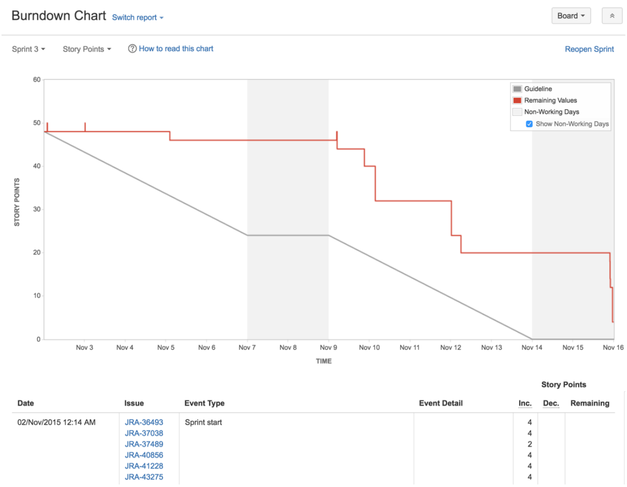
Our Product owner (Tochy) in collaboration with the team will decide which items make it into the sprint backlog. We will be using Jira scrum board to manage the sprint backlog.

14.2.5 Task Breakdown

The team will choose on which tasks to do voluntarily and based on their time availability for that sprint. The product owner will review the work overload and reassign if it is very unbalanced. Time spent on tasks will be documented individually and in comments on the story in Jira.

14.2.6 Sprint Burn Down Charts

any team member can generate the burndown chart from Jira.



				Story Points		
Date	Issue	Event Type	Event Detail	Inc.	Dec.	Remaining
02/Nov/2015 12:14 AM	JRA-36493 JRA-37038 JRA-37489 JRA-40856 JRA-41228 JRA-43275	Sprint start		4 4 2 4 4 4		

Figure 2: Example sprint burn down chart

14.2.7 Sprint Retrospective

The sprint retrospective will happen as a team and will happen in the friday meeting after the sprint is over. As a group we will document work done versus expected work, and as individuals we will document if it was too much work for our schedule or too little.

14.2.8 Individual Status Reports

Each member will report their status on a bi-weekly basis via discord. It will include their current progress on the work, if they will be able to finish their alloted work in the sprint, and if they need any help to overcome technical challenges.

14.2.9 Engineering Notebooks

The engineering notebook will be updated at least once a week and at least 1 page per 1 week interval unless their is a holiday or extended break for a team member communicated in advance. The team will show their notebooks to eachother weekly to make sure this happens. No witness signature is required.

14.3 Closeout Materials

14.3.1 System Prototype

We will have a fully functional database with all planned features and demo it to the customer with mock data. The customer will do Prototype Acceptance test and if they like it will take our implementation and create it with their own data instead of mock data for another test that we won't see the results of due to it being sensitive information.

14.3.2 Project Poster

We will use a white poster for our final presentation. We will include our brainstorming, major obstacles, major successes, and final product opinions on the poster board. It will be delivered during the final project demonstration

14.3.3 Web Page

Their will be a public project web page provided at closeout giving basic information about each team member, linking to the source code and demo video, and containing documentation links.

14.3.4 Demo Video

Our demo video will show the software being used with mock data, we are currently unsure of the video length but will be shooting for about 10 minutes to overview the entire application. We will cover the back end, front end and the ability to auto generate documents. We will not include B-reel footage.

14.3.5 Source Code

The source code will be maintained by either other cse classes or future senior ddesign groups that add features. The source code itself will be tured over to the pc provided by the senior design lab.

14.3.6 Installation On User PC

Currently unsure of how the customer will deploy software. We plan on setting up the system on a pc provided by senior design where they can do as they please afterwards.

14.3.7 User Manual

The user manuall will be a short and consice explanation of how the system should be used. It will be in the closeout material during submission.

REFERENCES

[1] Amazon. "Start Building on AWS Today". https://aws.amazon.com/, Accessed Sep. 23, 20121.

[2] Amazon. "Amazon Relational Database Service (RDS)". https://aws.amazon.com/rds/?p=pmc=dbz=3, Accessed Sep. 23, 2021.

[3] Arthur Haponik. "Text Extraction From Images Using Machine Learning". https://addepto.com/text-extraction-from-images-using-machine-learning/, Accessed Sep. 23, 20121.

[4] Django. "Django makes it easier to build better web apps more quickly and with less code.". https://www.djangoproject.com/, Accessed Sep. 23, 20121.

[5] Fishbowl. "The #1 Manufacturing and Warehouse Inventory Management Software". https://www.fishbowlinventory.com/, Accessed Sep. 23, 20121.

[6] React. "React A JavaScript library for building user interfaces". https://reactjs.org/, Accessed Sep. 23, 20121.

[7] Tableau. "Business Intelligence and Analytics Software". https://www.tableau.com/, Accessed Sep. 23, 20121.