

Marist College

Ski/Snowboard Shop Improvement

Team Rangers:

Colin MacDonald, James Vetro, Harrison Zheng, Jelani Joseph

CMPT 230L-111

Professor Zeppetelli

8 May 2020

Executive Summary

In summary, our client's initial issue was their way of storing data and operating the ski rental business: they would have to lug around a huge 3-ringed binder and write down all the customers information inside of it each time something was purchased. So after asking our clients some questions about how they want their new system to work and receiving the budget we were able to create a recommended course of action and design a new system that is entirely electronic. This new system would allow the clients to have the customers easily create a log of each item purchased, who purchased it, and at what day it was purchased. This data would then be sent to the database where an employee can easily look up who is in possession of what item and how long the customer is allowed to have the item before they must return it. The system also sends reminders to the customer when the due date is getting close just so there's no misunderstanding. On top of this our system would then be able to identify each object the store has for purchase with a barcode identification. Each item would have its own unique barcode specifying the type of item it is how much it would cost to buy a new one and how much the customer would have to pay if they damaged it beyond selling point. If the item were to break our system would also make it very easy to replace it since the manager would be able to directly access the supplier now with accurate information of how many new materials are in need. In conclusion, this system both makes it easier for the customer to buy goods and the employee's to do their job without having to store data into an unreliable source that cannot be backed up.

Table of Contents

Executive Summary	1
Table of Contents	2
Introduction	3
Summary of Relevant Facts	4
Problem	5
Analysis of Recommendations/System Proposal	6
Conclusion	9
References	10
Appendix	11
Figure 1 - Use Case Diagram	11
Use Case Write-Ups	12
Figure 2-1/Use case 1: Add new inventory	12
Figure 2-2/Use case 2: Reserve item online	12
Figure 2-3/Use case 3: Create customer account	13
Figure 2-4/Use case 4: Scan item at checkout	13
Figure 2-5/Use case 5: Return Rented Item	14
Data Flow Diagrams	15
Figure 3-1: Context Level Diagram	15
Figure 3-2: Level 0 Diagram	16
Figure 3-3: Level 1 Diagrams	17
Figure 4 - Entity-Relationship Diagram	19
Figure 5 - Hardware/software Design	19
Figure 6 - User Interface Designs	21
Figure 6-1: Example data spreadsheet	21
Figure 6-2: Rental Alert Email	21
Figure 6-3: Example Web Shop	21

Introduction

Running a ski rental business can be difficult especially when you have no way to track the equipment, but our team has a solution that could make this task seem trivial. Our clients have informed us of this issue and needed a way to make tracking their rented items much easier. This is a major problem because in the old system if an item was checked out there was no way to track it or tell if it was switched out for an identical item. So we made a new tracking system to accurately track each item and even create a starting date and due date for the item being bought. Each item will have a barcode on them which holds a unique identification number for that specific item. The number is saved and inputted into the system when it is delivered to the store. If anyone wants to check out the item they would purchase it normally. However, when the item is checked out, the date at which the item was bought is sent to the database and the item is now marked in the database as in-use. The company and customer can then check on the website to see when the item should be returned to the store, and the system calculates a late fee based on how many days the item goes past its due date.

With this system, the company can now easily track their equipment being rented out to the user, keep track of how many days late the user has kept the item pass due date and calculate an accurate price, and in general improve their overall security of goods.

Summary of Relevant Facts

Essentially, our system will be a virtual tracking and storage system that will make it much easier for the clients to keep track of the customer's information and equipment's information. We conducted an interview to organize the specifics of the system using these questions:

1. From a high level perspective, how is the current system used?
2. How do you intend the system to run?
3. How would you like the user to interact with the system?
4. How will you hold your customers accountable for rental damages and late fees?
5. Where will the customers be utilizing this function? Will it be used in the store or on the internet?
6. What is the budget for this project?

In return the clients told us their specifications:

- Current system is tracked by hand and time consuming no easy way to let customers know when due date has reached
- New system must use a barcode and when purchased rental information and user information must be saved to the computer
- The customer will be able to reserve the equipment they want to rent on our website and input their personal data
- There will be a security deposit on the equipment and if it's returned late we charge a late fee
- The customers will be able to reserve rental equipment online and pick it up in person when they arrive at the store. The system must also allow walk-ins for people who want to rent in person.
- Budget for project \$10,000

After obtaining the requirements from the user we were then able to begin looking at the big problem and crafting a solution to it.

Problem

The main problem we were able to derive from our clients is that they would like to change their system from physical paper storage to a virtual database. Instead of using a 3-ring binder to hold all their information on both the customer and the equipment, the client would like us to make a barcode tracker system for the equipment and store that information as well as the customer information into a large database in which they can easily search and locate their products.

Analysis of Recommendations/System Proposal

To solve the ski/snowboard shop's problem, we designed a comprehensive point-of-sale system that can handle all of the shop's day to day operations. Following the systems development life cycle, we took steps to analyze the problem and then design a solution for it. The first step of the analysis phase was to ask our clients several questions to find out what they expected to see in the new computerized system. These questions often revolved around the state of the system currently being used, potential opportunities for improvement, and specific features that the clients needed (see Summary of Relevant Facts for questions).

Next, we created a use case diagram to show typical interactions that will occur between the system and its users, which includes both customers and employees (see Figure 1). We also explained 5 of the most important use cases from our diagram in greater detail in individual write-ups (see Figures 2-1 to 2-5). The write-up in Figure 2-1, for example, highlights the event where a shop manager orders new equipment, and it includes many details such as what conditions must be satisfied prior to the start of an instance of the use case.

Our next step in the process was to take the information in our use case diagram and create data flow diagrams. Figure 3-1 shows the context level diagram. This diagram gives a high-level view of different users and how they will interact with the system. It only provides basic information, but it gives context on the entire system. The level 1 diagrams in Figure 3-3 have much more detail. We took the 5 most important use cases from earlier and converted them into individual data flow diagrams. This allows us to see what types of data and which databases are needed to complete a task. These diagrams were then merged together into a single level 0 data flow diagram (see Figure 3-2). This helps to show how the data from each system interacts with other systems. It also allows us to identify the common database types that would be needed. Everything was able to work with three databases: data about inventory, a customer database, and data about rentals.

In the last step of the analysis phase, we created an entity-relationship diagram, which can be seen in Figure 4. This is a diagram which lists different entities in the system, what attributes and keys they have, and their relations to each other in terms of primary/foreign keys and possible quantities of relationships. The entities are customer, order, product, employee, and vendor and they each play an important role in the ski shop's operations. When it comes time to actually build the databases, this diagram will be very useful in determining the fields for each table.

We began the design phase of the systems development life cycle by putting together a list of hardware and software specifications. We had to balance the need for

reliable equipment with the fact that the ski shop is a small business but its volume of customers varies depending on the season as well as day of the week. After much research, we arrived at the following prices for hardware: POS price: \$599 each, web server: \$2,000, application server: \$300, database server: \$786. The full specifications can be found in Figure 5. This means that the initial cost of implementing our system will start at \$3,685, which is well within our clients' \$10,000 budget.

The final step we took before being ready to implement everything was creating some user interface designs, which are shown in figure 6. Figure 6-1 shows the interface a manager could use to directly edit the details about existing items in inventory, or to add new items. Figure 6-2 shows an email that could be sent to customers when their rental is close to being due. A similar message can be sent with a text message. Lastly, figure 6-3 shows the design of the landing page of the online store where customers can buy items or reserve items for rental. This would be the only interface a customer would need to interact with because everything else is done in the backend or by an employee.

In our system, a typical client will be anyone viewing the shop's website in a web browser on their laptop or desktop. A customer will also access this website which is hosted by a web server. An in-store POS terminal that scans purchases is the other type of client. The terminals come with software from Epos Now when purchased. The application server will handle the system's business logic, such as running the online store. Finally, a database server will store all important records, such as the shop's inventory and customer data.

Since the system will be using multiple servers with designated tasks, it is an example of n-tiered architecture. This setup was chosen because it is the most flexible design. If one specific process is being overloaded, which could happen in the winter when ski and snowboard rentals peak, the server that the process is running on can be upgraded, or another server can be added. This allows for easy load balancing. Also, the client-server architecture allows multiple customers and their web clients to communicate with the same information that the POS terminals access, allowing information storage on a database to be centralised.

Each server will cost money, as well as the POS terminals and POS software. Most of the server processes can be managed using open source or free tools. However, paid POS software was chosen for their superior quality, the ability to receive quick support, and integrations with credit card providers.

Implementation of our system is a straightforward process. Our clients will choose a day that works for them, and we will make sure that all of the POS terminals and servers arrive that day along with professionals who will set it all up. The terminals will be placed on the front counter where customers go to rent and return equipment. The servers will be kept at the back of the shop in a room that only the top managers

have access to since the servers will store sensitive business data. If everything goes according to plan (i.e. no defective equipment), the terminals and servers should be up and running within a day.

Once the new system is set up, employee training will take place. Professionals who have experience using the system will first walk the shop managers through the basics of using Epos Now's software on the POS terminals. Then they will show them more advanced functionalities like changing an item's price for a limited-time deal and collecting data on the shop's sales within a certain period. Once the managers become familiar with the software, they will teach their employees how to complete a transaction, process returns, register new equipment, etc. Due to the intuitive design of the interface on the terminals, employees will know how to use them to efficiently complete tasks in just one week of training. The POS terminals also come with help guides and manuals that employees can refer to at any time if they are having trouble using the software.

Conclusion

By implementing our recommended course of action the clients will be able to save a mass amount of money in the near future. Due to the system tracking when the original item was first purchased and how long the customer has kept the item will allow the business to accurately charge late fees and save them the time of having to hand write the information in a book that could potentially be lost, and is much more tiresome to update. This will also ensure their system is up to date and more user-friendly which will encourage customers to buy products from the store again. So our course of action would solve the clients problem of keeping all of their information in a book to having the information be stored in a database, allow customers to have the ability to access the stores goods from the website and pick the item up when ready, and help the employees search for specific data with ease.

References

Works Cited

Dennis, Alan, et al. *Systems Analysis and Design*. 7th ed., e-book, Hoboken, Wiley, 2019.

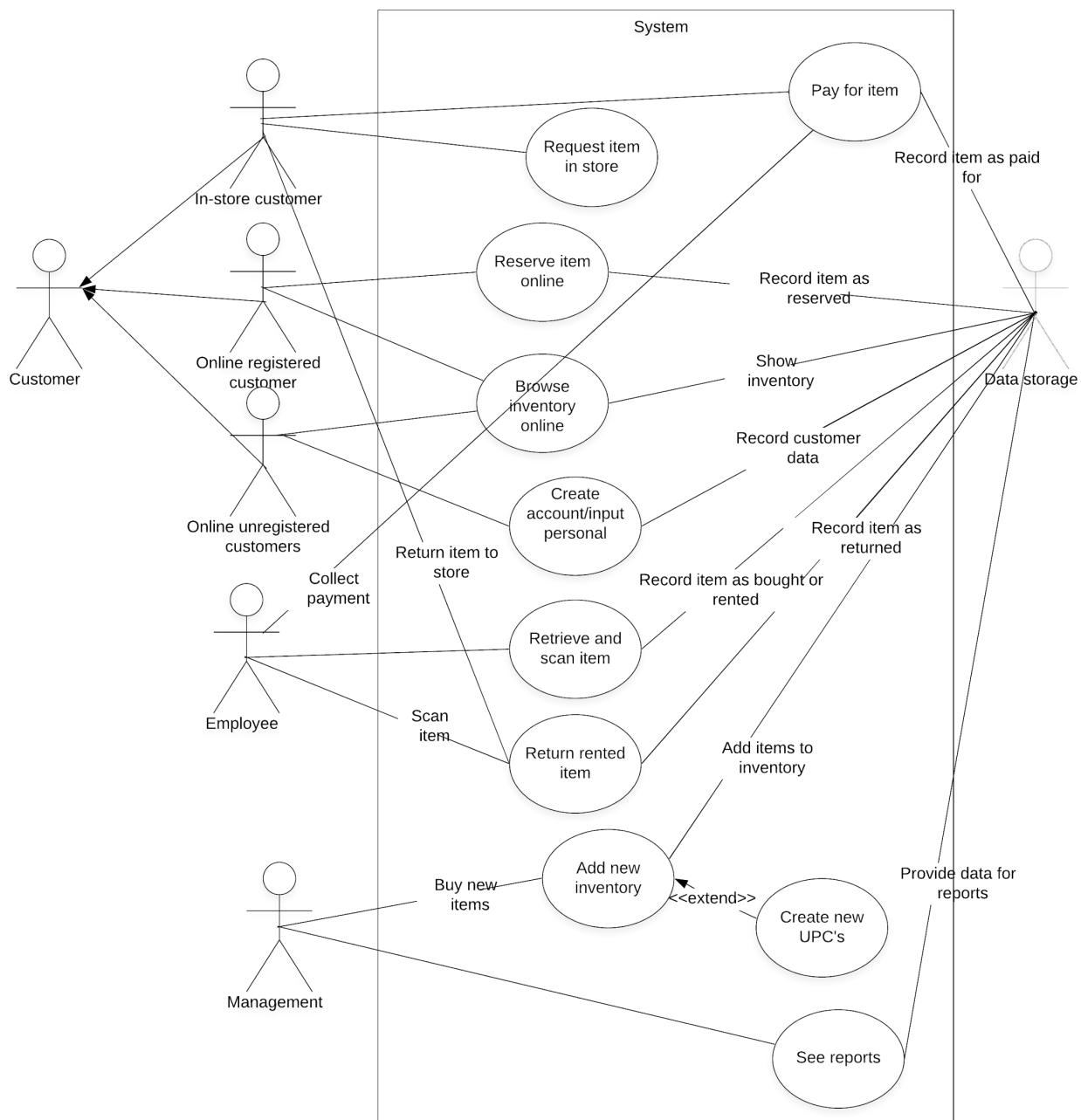
"Epos Now Systems." *Epos Now*, www.eposnow.com/us/systems. Accessed 8 May 2020.

Lahn, Mark. "How much does a server cost for a small business?" *ServerMania*, 22 Apr. 2019, www.servermania.com/kb/articles/how-much-does-a-server-cost-for-a-small-business/. Accessed 8 May 2020.

"MLA FORMAT IN DETAIL." *Online Writing Lab*, Aims Community College, 2018, www.aims.edu/student/online-writing-lab/common-formatting/mla-format/format-in-detail.php. Accessed 8 May 2020.

Appendix

Figure 1 - Use Case Diagram



Use Case Write-Ups

Figure 2-1/Use case 1: Add new inventory

Use Case name: Add new inventory	ID: 1
Actor: Manager	
Description: Manager purchases new equipment from supplier in order to restock inventory	
Trigger: Manager purchases items for new inventory	
Type: External	
Preconditions: Inventory needs restocking.	
Normal course:	
1. Manager purchases new equipment.	
1. Data about items is entered into system (price, quantity, image, description, UPC)	
2. UPC tags for each item are created and attached.	
3. Information is published to website.	
4. Employees and customers are notified (email/text message) about new inventory and deals.	
Post conditions: New items in inventory and website, new UPC codes on record.	

Figure 2-2/Use case 2: Reserve item online

Use Case name: Reserve item online	ID: 2
Actor: Customer	
Description: Customer completes an online order to pick up in store.	
Trigger: Customer completes an online order.	
Type: External	
Preconditions: Customer has account, equipment they have requested is available in the database.	
Normal course:	
1. Customer logs in and browses inventory, finds equipment they want.	

2.	Customer clicks on rent item button.
3.	Customer fills out online form with when they want to pick it up and how long they want it.
4.	Rental request data is entered into system.
5.	UPC is marked as reserved.
6.	Customer picks up and pays at date they entered.
Post conditions: Equipment is paid for and rented, system removes items from available inventory.	

Figure 2-3/Use case 3: Create customer account

Use Case name: Create customer account	ID: 3
Actor: Customer	
Description: Customer creates an account and it gets saved to the store's database	
Trigger: Customer goes online and clicks "create new account"	
Type: External	
Preconditions: Customer does not already have an account	
Normal course: <ol style="list-style-type: none"> 1. Customer clicks "create new account" 2. Customer fills in required fields (name, email, password, phone number, etc.) 3. Email is sent to customer for verification. 4. Account details are stored 	
Post conditions: Account is created and stored in database, notify customers via email that their account has been created, periodically send customers deals and promotions	

Figure 2-4/Use case 4: Scan item at checkout

Use Case name: Scan item at checkout	ID: 4
Actor: Employee	
Description: Employee scans item at checkout to total up the cost for a customer.	
Trigger: Customer brings item to register to rent or purchase.	
Type: External	

Preconditions: Customer wants to buy or rent an item. Item shows as available in database and is currently not in use.
Normal course: <ol style="list-style-type: none"> 1. Customer places item on counter. 2. Employee scans all item UPC's into POS system. 3. Customer provides any coupons or membership info. 4. POS totals up the prices. 5. Customer pays for items. 6. Items are recorded as sold.
Post conditions: Payment collected, items removed from inventory Data of item is searched for in database and price is returned, it is then marked as sold, and specific item is marked as in use.

Figure 2-5/Use case 5: Return Rented Item

Use Case name: Return Rented Item	ID: 5
Actor: Customer	
Description: Customer brings rentals back to store and an employee scans them in	
Trigger: Customer brings item back to store	
Type: External	
Preconditions: Customer wants to return item, or has run out of time in rental period, item in database shows as expired or in use for too long.	
Normal course: <ol style="list-style-type: none"> 1. Customer returns the item back to the store. 2. Employee scans the item, and checks it for damage. <ol style="list-style-type: none"> a. If there is no damage, the item is re-entered into the system. b. If there is damage, the item is marked as damaged and the customer is fined. 3. UPC code is marked as available. 4. Item is returned to storage. 	
Post conditions: Condition of item is checked by shop worker, if item is in proper condition then it is added back into the store's database, if item is in poor condition then it is removed from database and customer is fined. The item is cleared from the customer's account, the item is re-entered into the system and stored.	

Data Flow Diagrams

Figure 3-1: Context Level Diagram

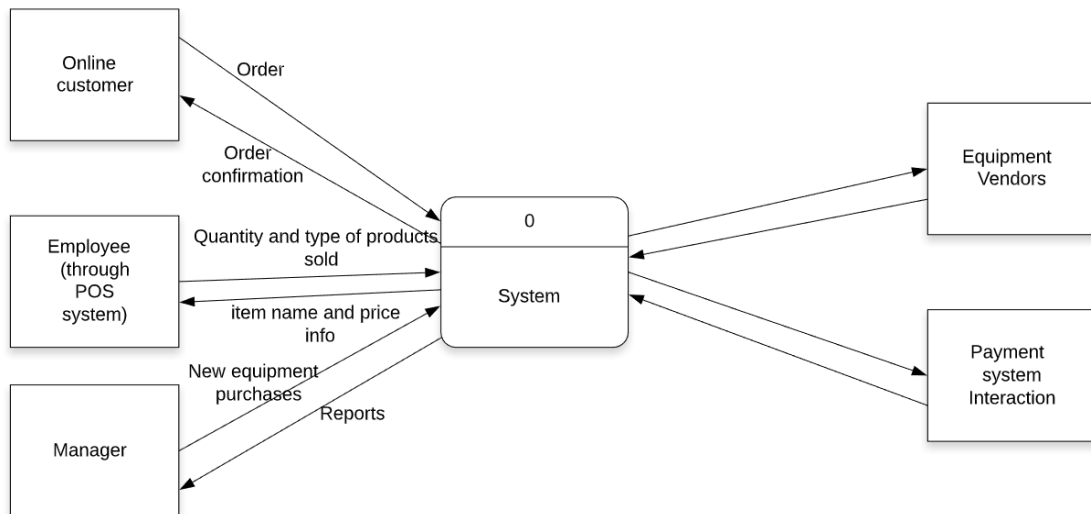


Figure 3-2: Level 0 Diagram

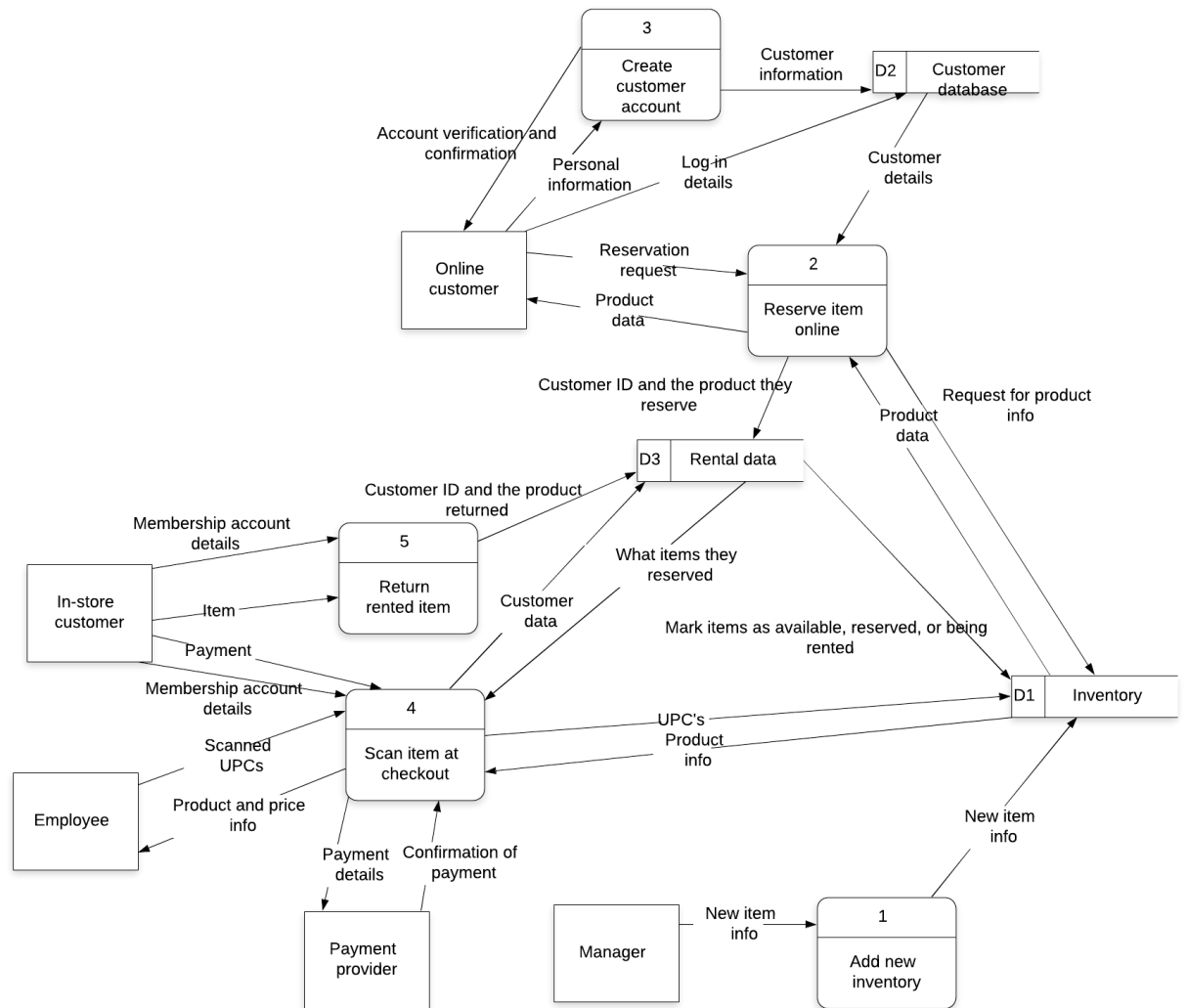
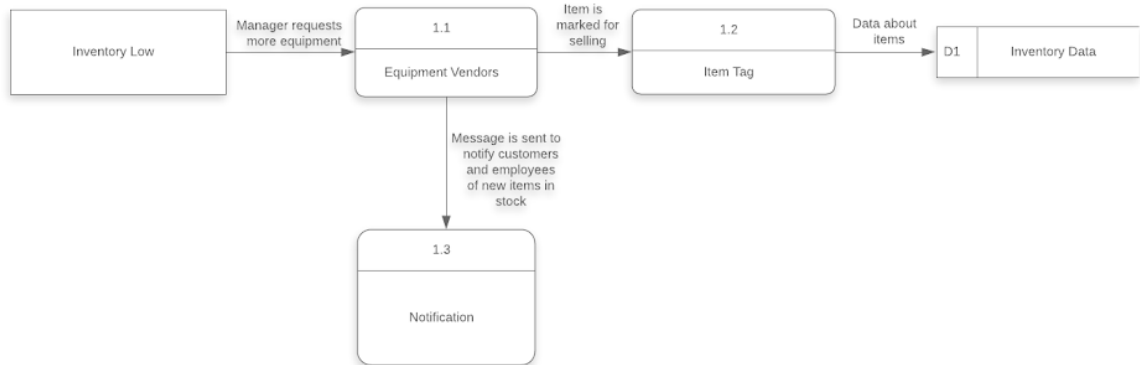
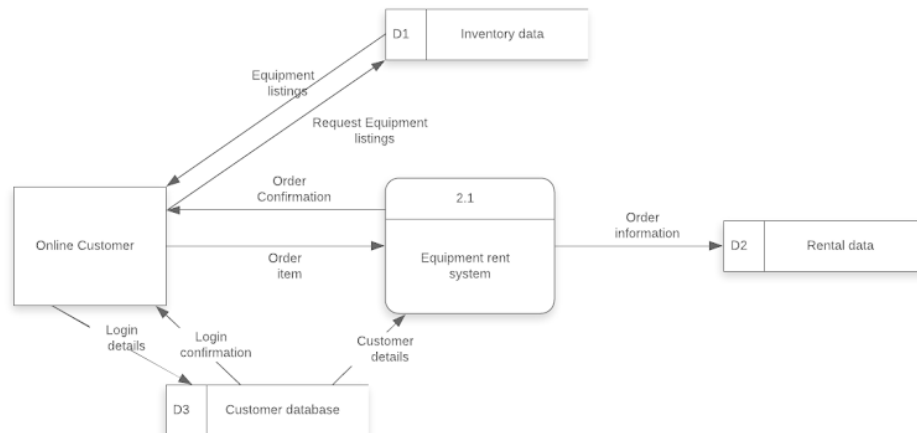


Figure 3-3: Level 1 Diagrams

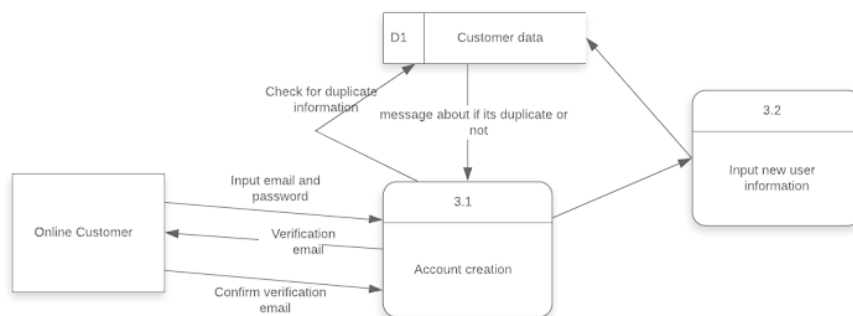
1. Add new inventory



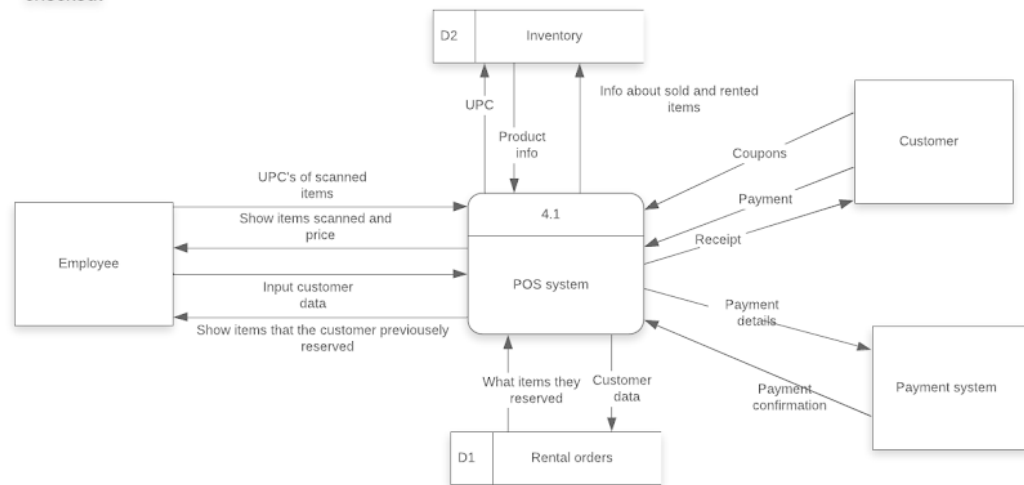
2. Reserve item online



3. Create customer account



4. Scan item at checkout



5. Return Rented Item

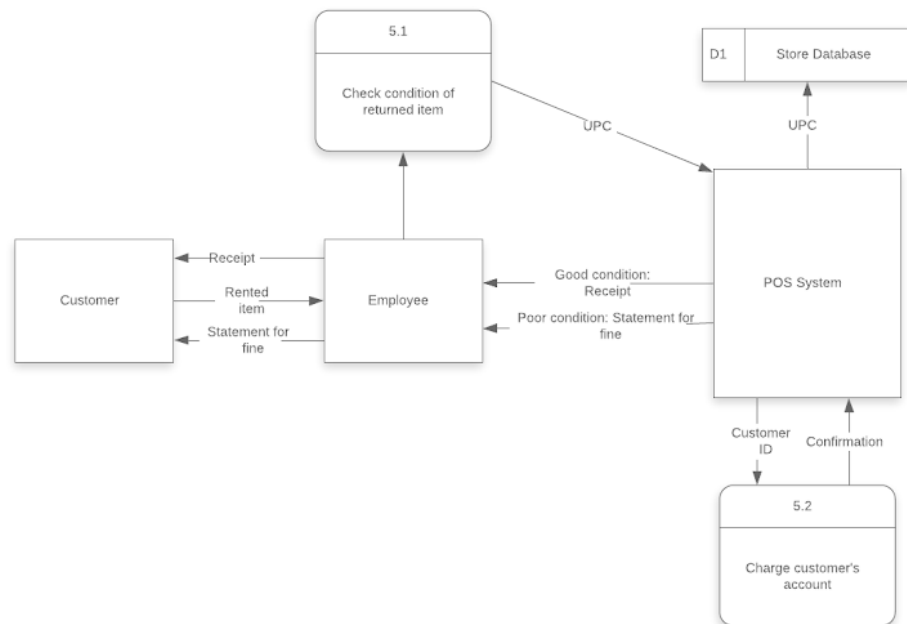


Figure 4 - Entity-Relationship Diagram

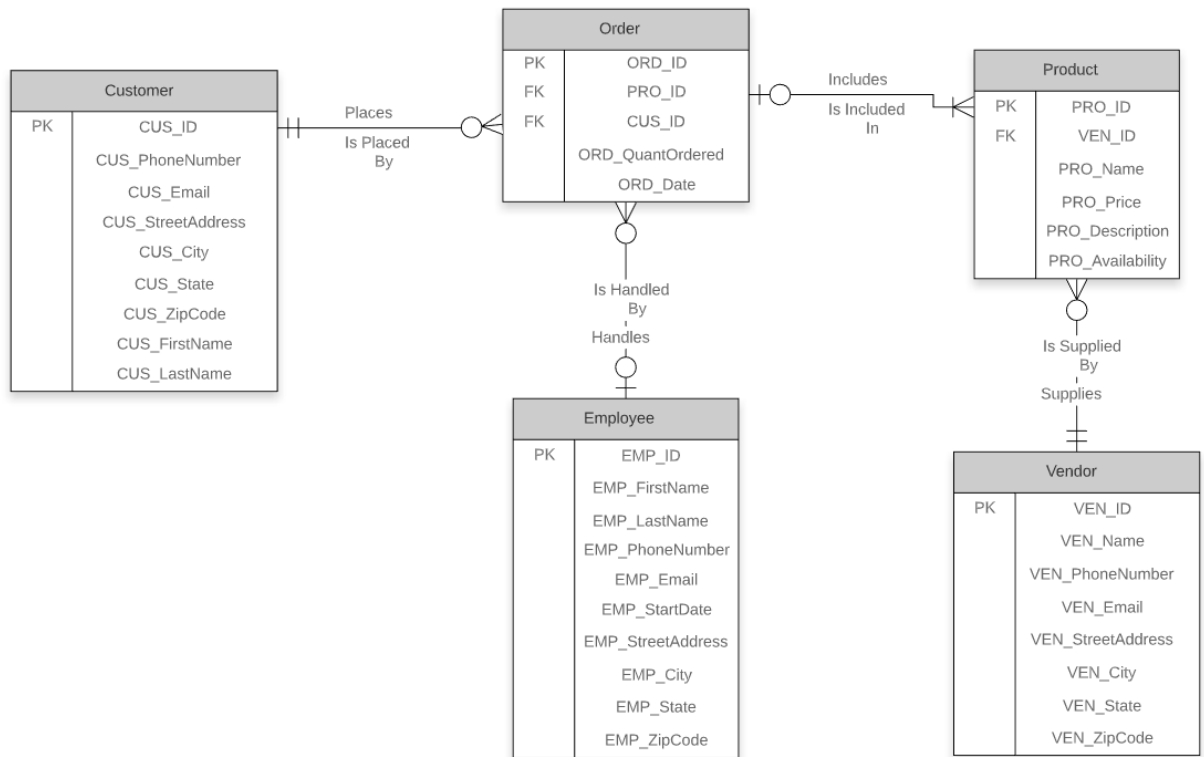


Figure 5 - Hardware/Software Design

	In-store POS Terminal	Standard Web Server	Standard Application Server	Standard Database Server
Operating System	Linux	Linux	Linux	Linux
Special Software	Epos Now	Nginx	Java	Oracle
Hardware	Full system POS (from contractor Epos Now) (includes miscellaneous equipment)	4 TB disk Drive Intel Xeon E3-1225 v5	4 TB disk Drive Intel Xeon E3-1225 v5	8 TB Disk drive Raid INTEL® XEON® SILVER 4215 PROCESSOR
Network	10 mbps Ethernet	Dual 100 mbps ethernet	Dual 100mbps ethernet	Dual 100 mbps ethernet

Figure 6 - User Interface Designs

Figure 6-1: Example data spreadsheet

Item Code	Item Type	Item Brand	Item Model	Customer name	Customer Address	Customer phone	Payment info	Date Rented	Date Due
5151272	Ski Pair	SkiMan	Model X	Marsha K	7 Sou St	(191) 282-1818	Credit Card	12/12/19	12/19/19
6182901	Snowboard	BoardKing	Model 2	John S	78 Un St	(588) 183-7592	Check	12/13/19	12/20/19
1829182	Ski Poles	SkiMan	Model Y	Reginald E	18 Yi St	(818) 756-1238	Credit Card	12/13/19	12/20/19
3189281	Ski Goggles	BoardKing	Model 1	Bob G	21 Ska Rd	(381) 282-1818	Paypal	12/14/19	12/21/19
2993839	Ski Pair	SkiMan	Model X	Harvey B	84 Ska Rd	(381) 282-1818	Credit Card	12/15/19	12/22/19

Figure 6-2: Rental Alert Email

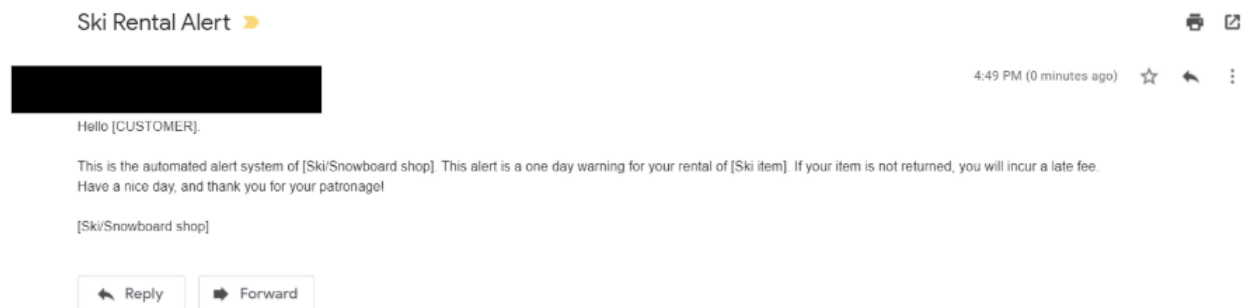


Figure 6-3: Example Web Shop

