**S**ervice.**I**nformation.**M**anagement

Application WareHouse

Final Case Report

-Shashank M

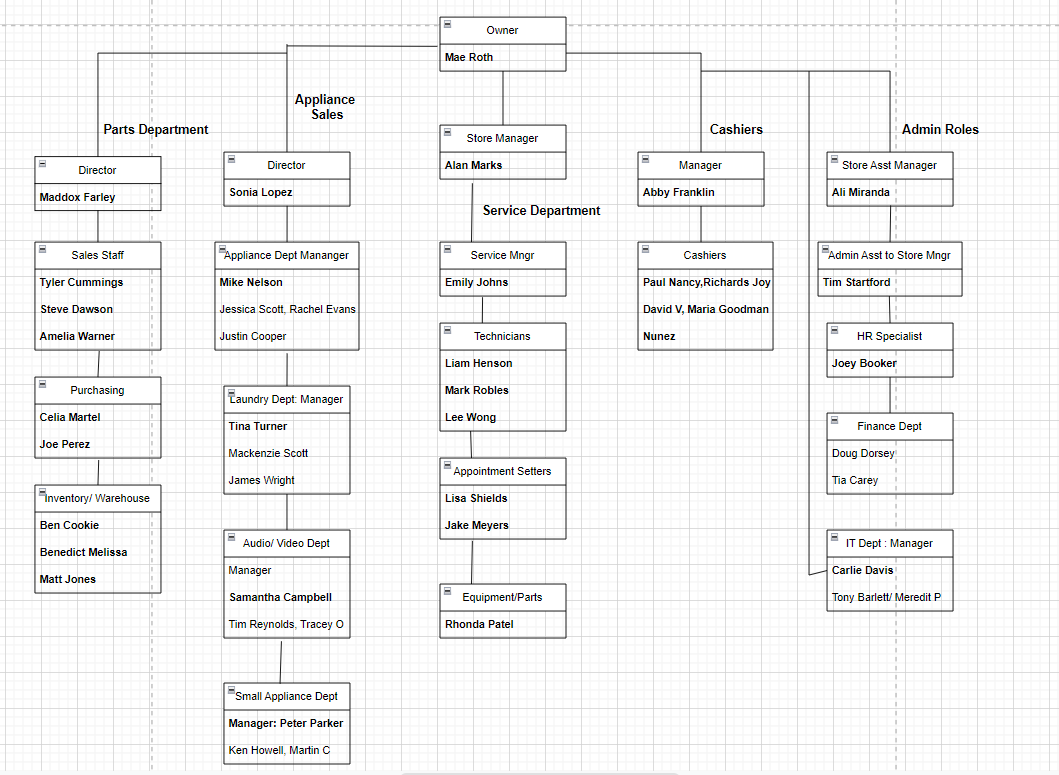
Index

Changes made are in *purple.*

* Organizational Chart
* SWOT Analysis
* Mission Statement
* Opportunity Statement
* WBS for Analysis Industry
* Risks for Service Department
* Questionnaire Open/Ended
* Data Flow Diagram
* Context Diagram
* Decision Table
* Decision Tree
* Use Case Diagram
* Object Model
* Instance Examples
* Software Choice
* Hierarchical Diagram
* Screen Examples
* Prototyping
* Report Types
* ERD Diagram
* DataBase Design
* Sample Data/Insertion
* ERP Softwares
* SIM: Network
* Security

**Organizational Chart** :

Made changes to ORG Chart, and updated with all teams, staff.



**SWOT** Analysis:

**SWOT** stands for Strengths, Weaknesses, Opportunities, and Threats, and so a SWOT analysis is a technique for assessing these four aspects of your business. SWOT Analysis is a tool that can help you to analyze what your company does best now, and to devise a successful strategy for the future.

Looking at the Warehouse case, Strengths and Weaknesses are mostly internal like faulty system error, or a malfunction or inventory fault. And we can also change our team/staff with properties. When it comes to Opportunities or Threats it is the opposite, they are quite external. Examples include, competitors, price of products(raw), and rate of customers shopping trends

By analyzing the Warehouse Website:

**S**: Branded site, 5 -star service, faster delivery speeds, low priced products.

**W**: Issues with the website, not strong mission statement, not enough browsing/pages to see.

**O**: Can generate revenue with the services with the home appliance parts, and user friendly UI to seem user friendly to customers who use the website. Creating 2 services departments as it said in the email to generate products/services. We can add more pages to website like Contact US Page to get feedback from customers, add appointment setters, technicians page so they can explore. We can add a delivery packaging system and tracking system so the customers know where their package is, and when it will arrive.

**T**: There are rival companies who may pay more, better products with reviews and all these factors to consider. Not enough salary for the workers because of competitive hire from rival companies that pay higher with well reputed businesses, demanding hours due to rising product lines etc.

**Mission Statement** :

AW’s Mission Statements states “*Our Mission, We are your one stop shop for your home appliances and parts*”.

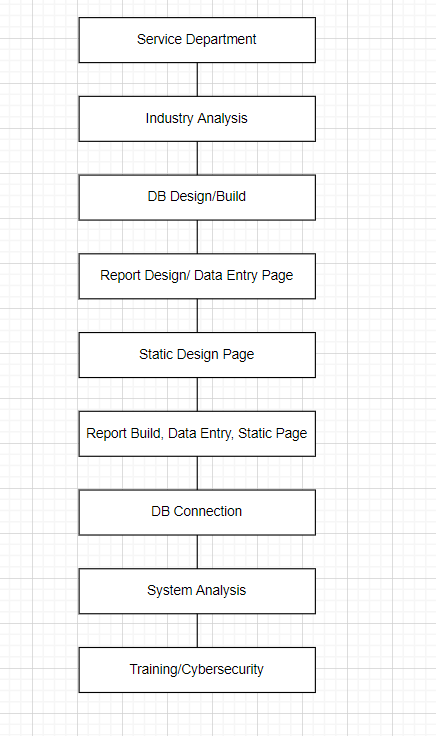
My honest opinion is it does look good in a certain way, but it does not catch the reader's attention, more like it doesn't act like a hook when reading an essay. For example Nordstorm mission statement states “To give customers the most compelling shopping experience possible”, it gives out positive vibes, the fix would be appealing to customers with warmth and in a pleasing way would do a lot of better. The previous statement says we are a stop for shop for your parts, but that's all it says. To fix the statement from the original would be better as “ Our Mission, to provide safe, secure, comfortable, reliable home appliances, so we are a part of your family. We are a value, feedback driven business running to believe in strong and homecoming familial bonds. ”.

**Opportunity Statement**:

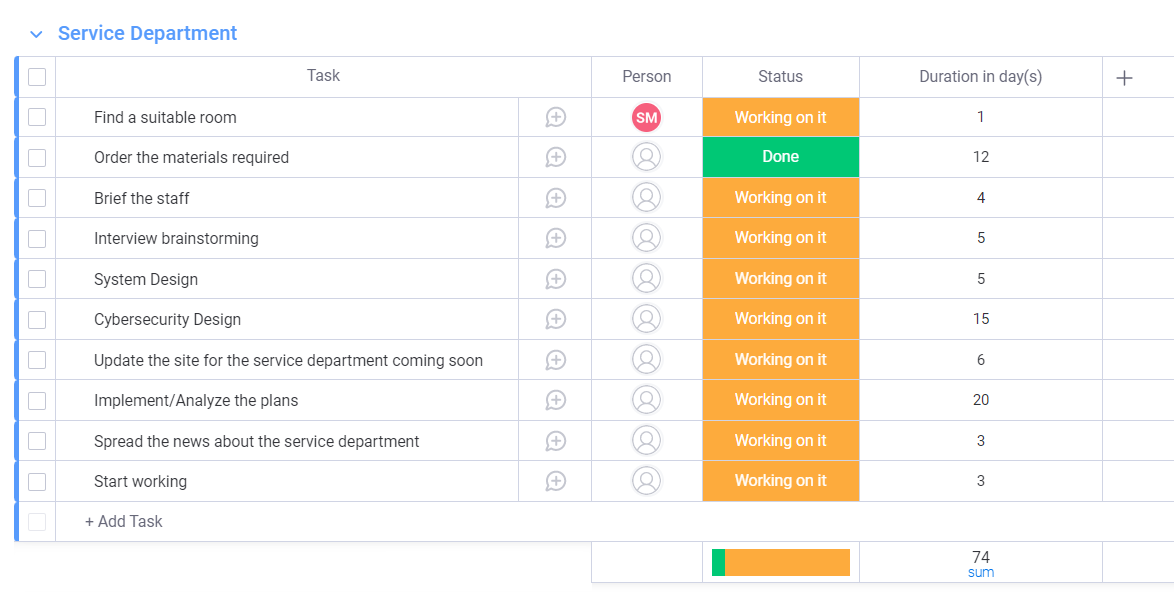
**Current State:** Mae has decided to implement two departments to generate revenue using the new departments and use the old for spare usage.

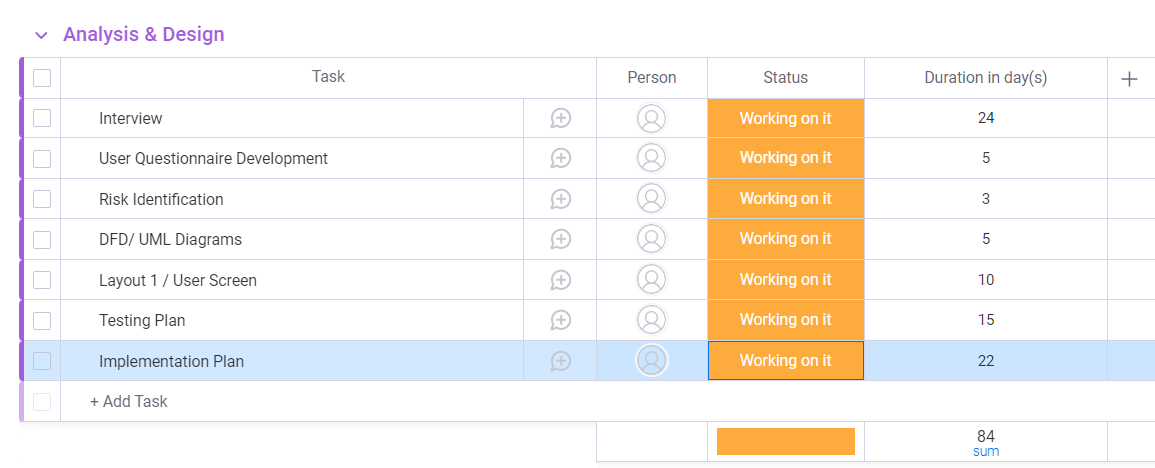
**After effect**: The company might lose the revenue in after sales, the 3rd party vendors often replace original with the spared parts to generate revenue. I believe that this led Mae Roth to develop the Service Appliance department.  
 **Needed State**: Mae Roth wants to provide 5 star service to all the customers who visit the AW, wants to introduce the new features that will come to AW, The new dept is set to increase customer satisfaction and customers as well as profit.

**WBS for Industry/System Analysis** : Added the WBS info in a Project Planning template, and made WBS using the resource template



Below is the project plan in a project planning software (monday.com): Assuming 8 hours/ day to work.





**Risks for the Service Department System**: Made changes to the risks and added new risks

1. When installing the service department, there are **scheduling conflicts**.
   1. To solve this we can make a master schedule, budget for breaks, and we can hire some people for back-ups, and give workers the ability to propose an alternate time to work, this way the service department should hold with timings.
2. IT Sector: If there are any problems, we can hire technicians beforehand and help us to solve IT problems or any any software related problems, before any customers are in the service department.
3. Financial Sector: We need to make a fixed amount to allot for the service department, it might fluctuate a little bit, but we need to make a fixed size so we don't run on empty while constructing a room and equipment, software workers for the Service Department. Make an excel sheet about all the requirements needed to make the department and share it with all the workers.
4. Roles: The workers need to know what their role is in the service department, if they have any questions, concerns, doubts, it's best to approach the floor supervisor and they can help proceed with the next steps

**Questionnaire** : **Open Ended/ Close Ended**

To **Melinda Parks** (Programmer):

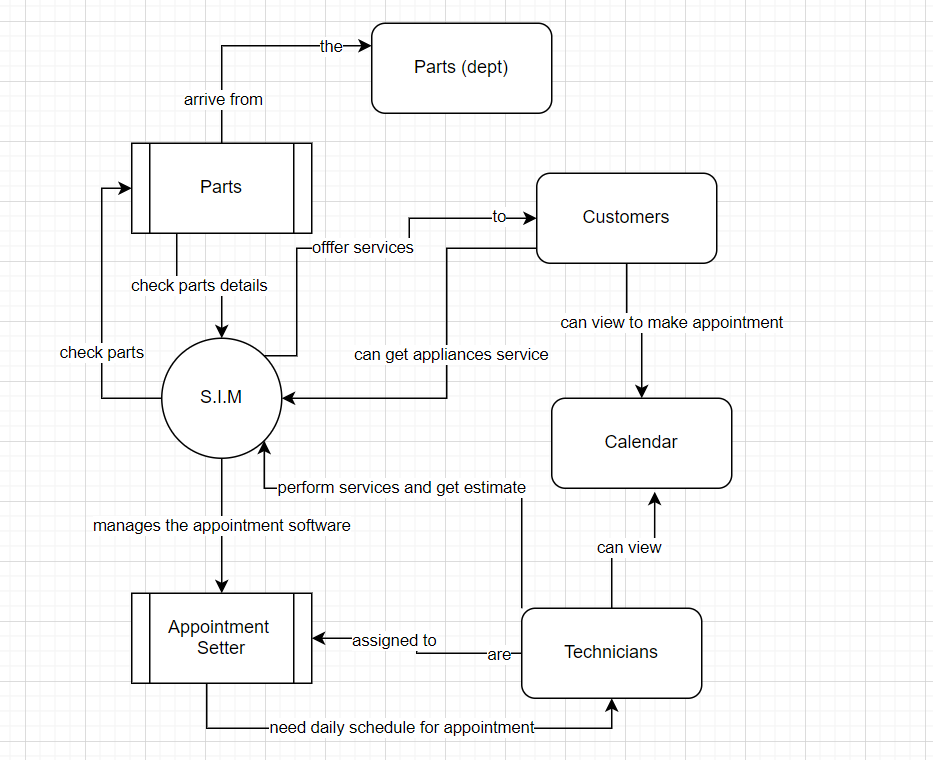
1. What are your thoughts on using Agile development techniques in this environment?
2. Are you leaning more towards Agile or other dev techniques?
3. What requirements are you expecting from a programmer in this position?
4. What do you know about this programming job, the skill it requires, and the maintenance it requires?
5. What coding languages are you proficient in? And why?
6. How do you keep up with current trends/tech in this field?
7. What is your ideal work environment?
8. What are you expecting from me to program for the service department?
9. What features are needed to add to the department?
10. What are the purpose of those features?
11. How does the UI need to look for this department? Any requirements do I need to meet?

To Emily Johns (Service Manager)

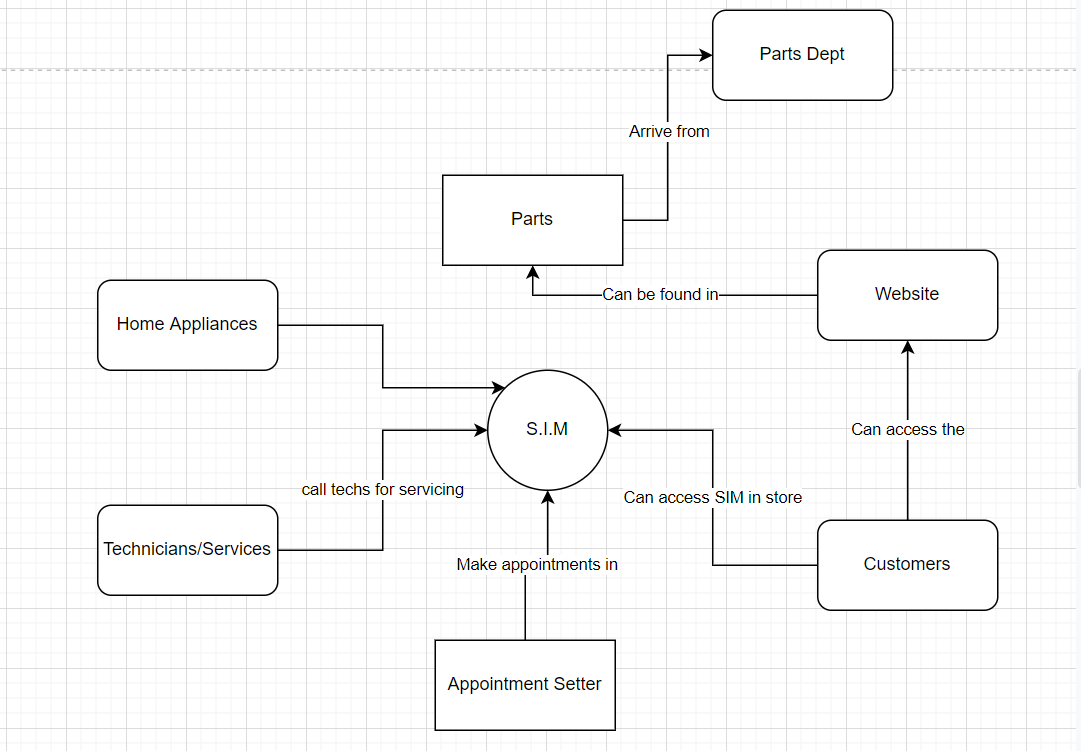
1. How many service requests can we expect in a day?
   1. What types of requests? How many do we get in a day? Average?
   2. What factors or external factors will affect this position?
2. What types of metrics and data can we provide?
   1. How can we track it? What type of software will be used to track?
3. What does a service manager do?
   1. What are the methods in this service request?
   2. Where does the service request start from?
   3. Where will this manager be helpful in the service request position?
   4. What do I need to know about this position?
   5. If there is a bad experience with a customer regarding service how can I proceed next?

**Sampling**: I think it’s best to go with stratified sampling, because there are advantages to using this technique, the main is it captures key population in this interview questions we want to interview for like the programmer who designs the interface and the Service Manager who will be overlooking the department. Similar to a weighted average like picking randomly, this method of sampling produces characteristics in the sample that are proportional to the overall population. Stratified random sampling works well for populations with a variety of attributes but is otherwise ineffective if subgroups cannot be formed.

**Data Flow Diagram**: Added active verb nouns to arrows, and customers access the website, and parts arrive from the parts store.



**Context Diagram**: Added active verb nouns to arrows, and customers access the website, and parts arrive from the parts store.



**Decision Table**: Removed excessive “Yes” and added (-), and updated it to a single yes, and multiple no’s.

| Purchased Appliance | Bought Service Plan? | Serviced more than 3 times? | Service Fee? | Discount on the service fee? |
| --- | --- | --- | --- | --- |
| Yes | X | X | X | X |
| No | No | X | Yes | No |
| Yes | - | No | No | No |
| No | No | Yes | - | Yes |

**Decision Tree**:

**Conditions:**

Appliance purchased from AW

Service plan bought by customers

Appliance repaired more or equal to 3 times in the year

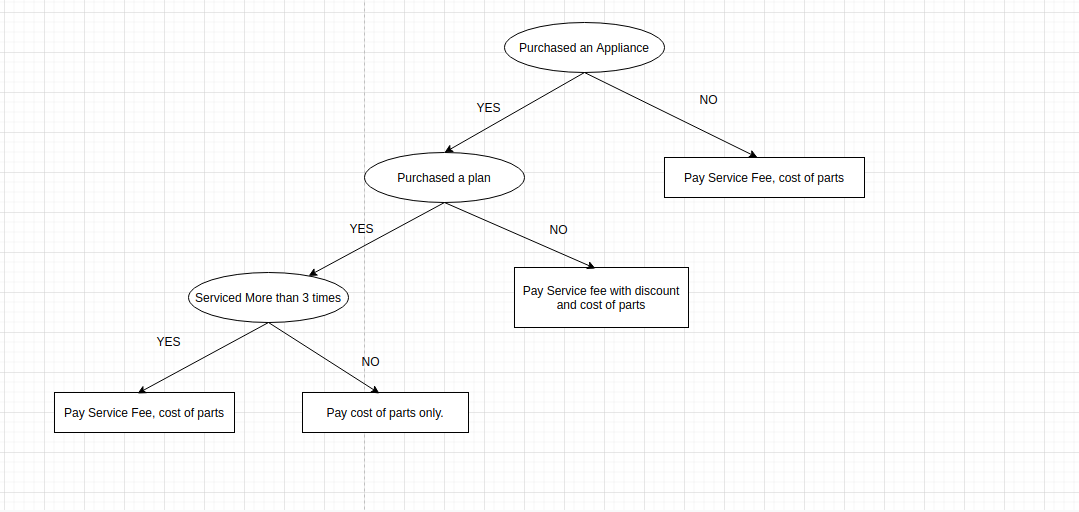
**Actions:**

No service plan

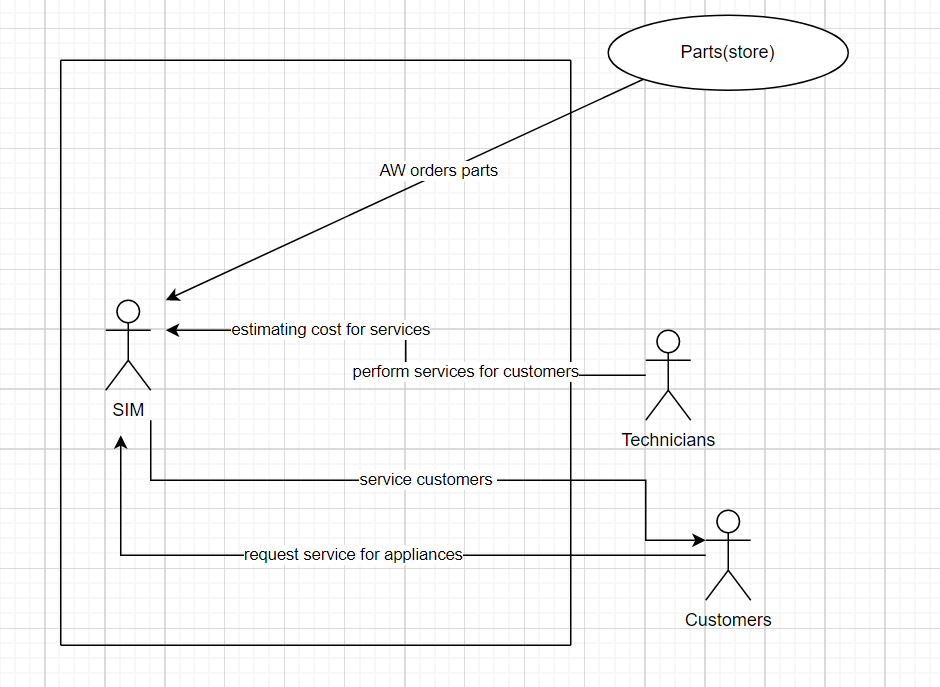
Customer paid the cost of parts

Paid the full service plan

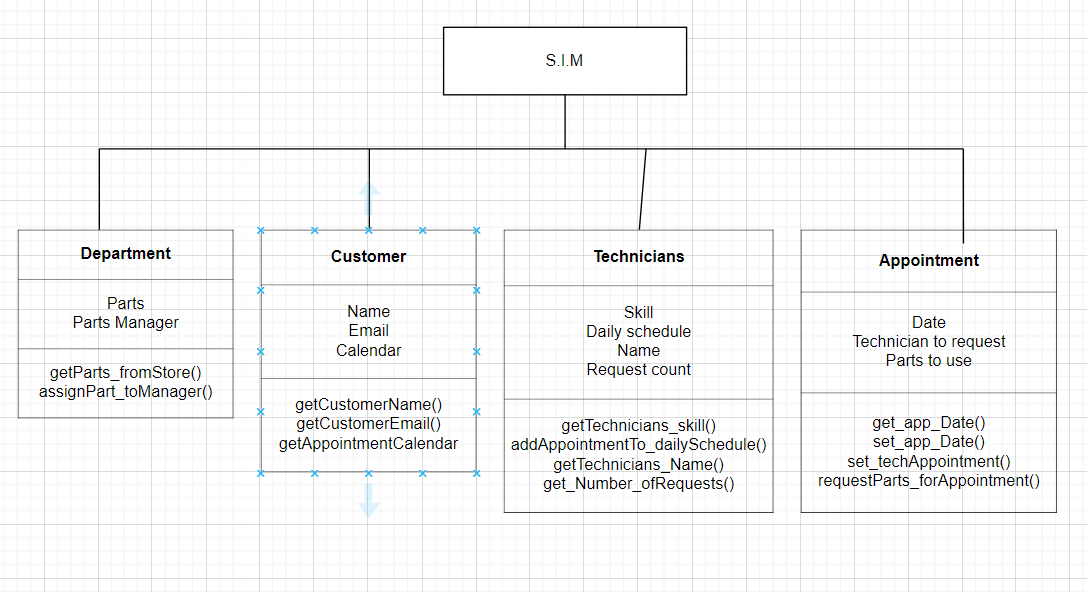
Customer gets 20% off discount



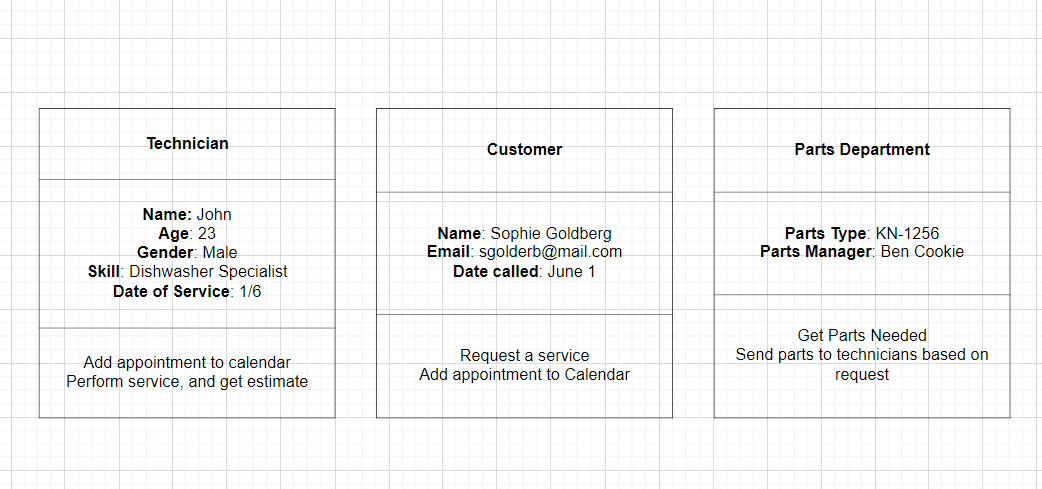
**Use Case Diagram**: Actors are outside the boundary, and made changes where the processes scope is smaller.



**Object Model**: Made changes to the object model with improved attributes and methods.

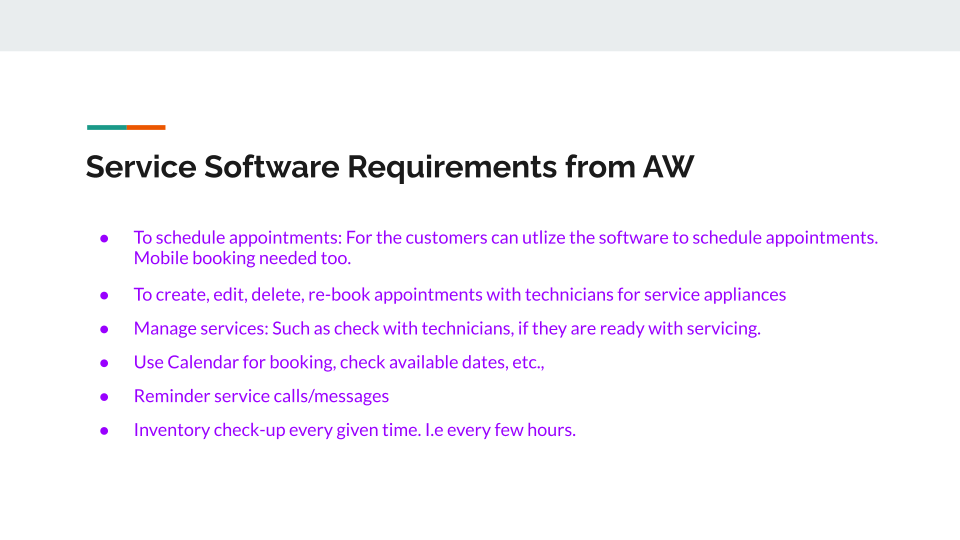


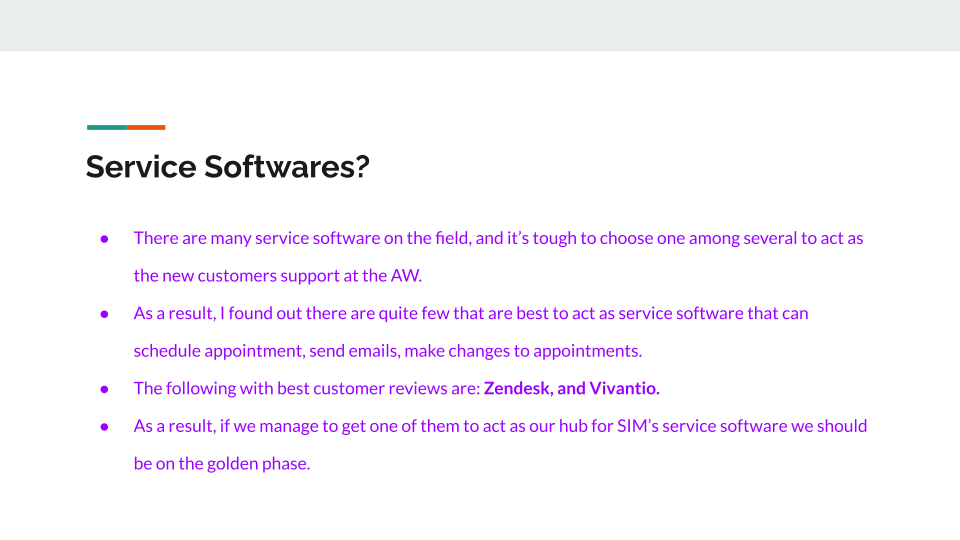
**Instance Examples** : Made changes to Instance examples and used entities from Object-Modeling.

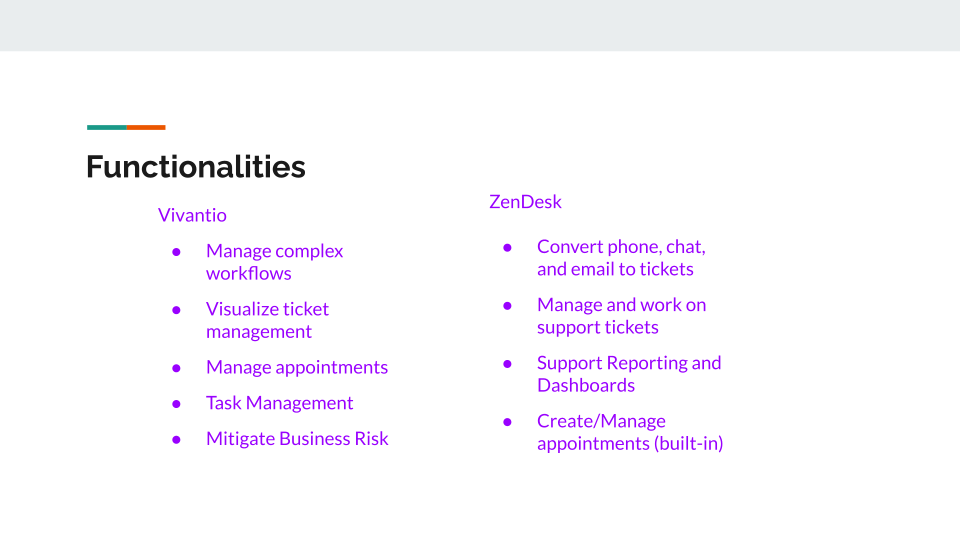


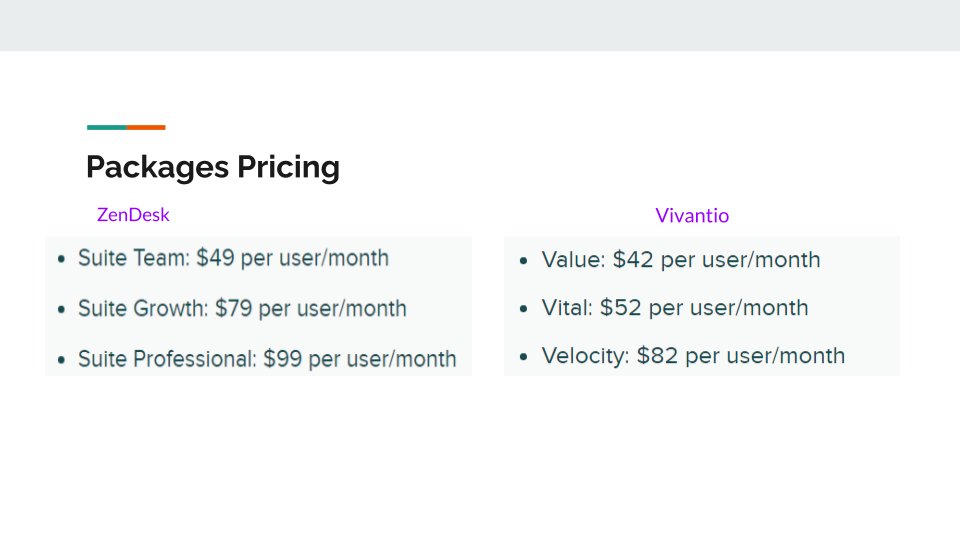
**Service Software**: Made changes to the slides, added Service Software requirements, build costs, nice-to-have list with additional functionalities made changes in ***slides*** as well.

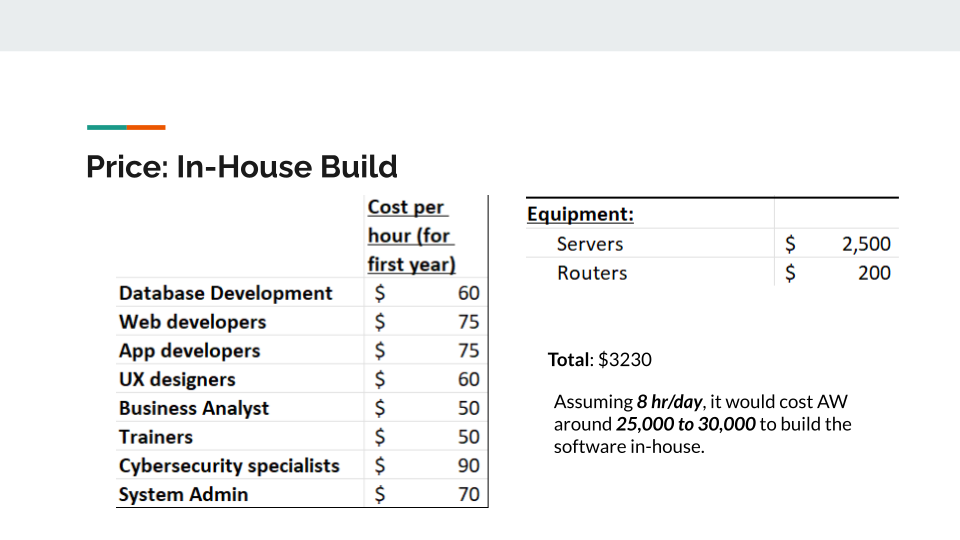


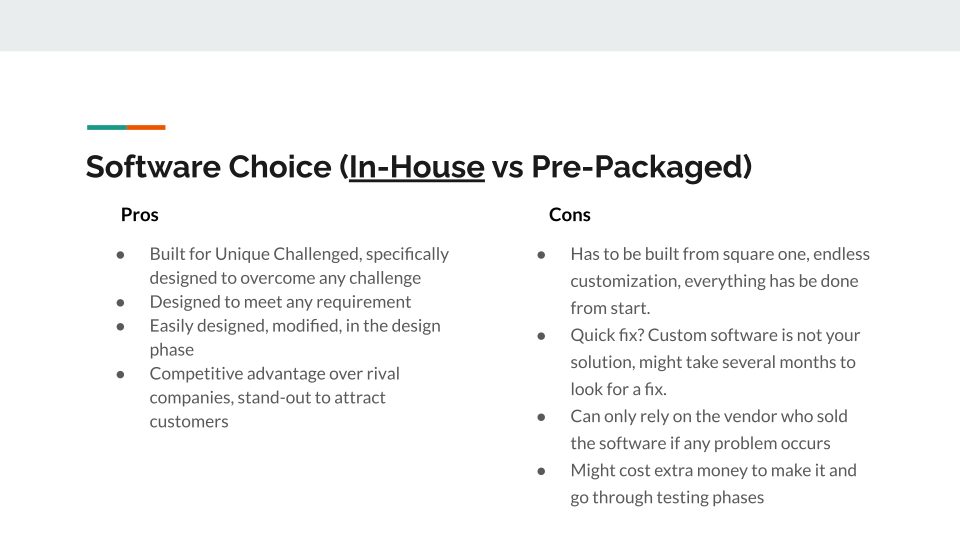


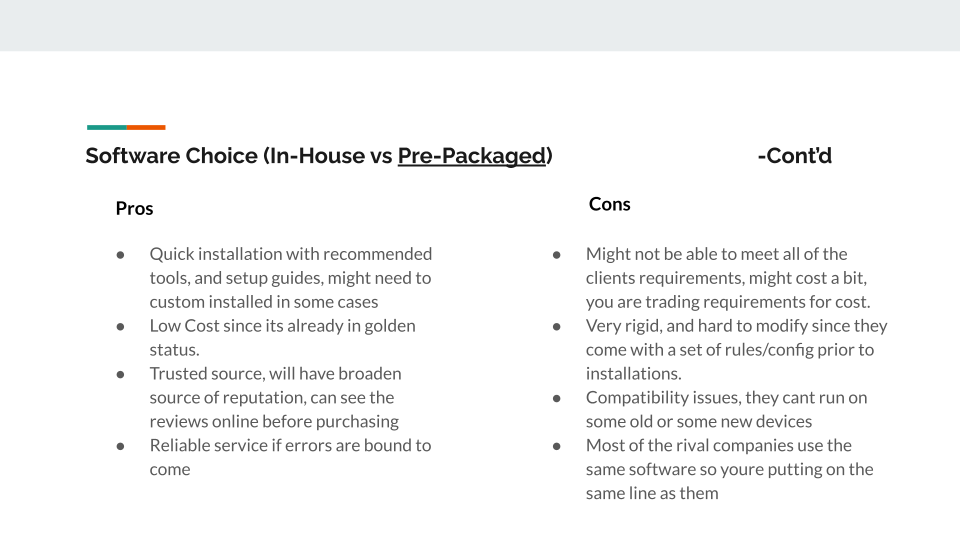


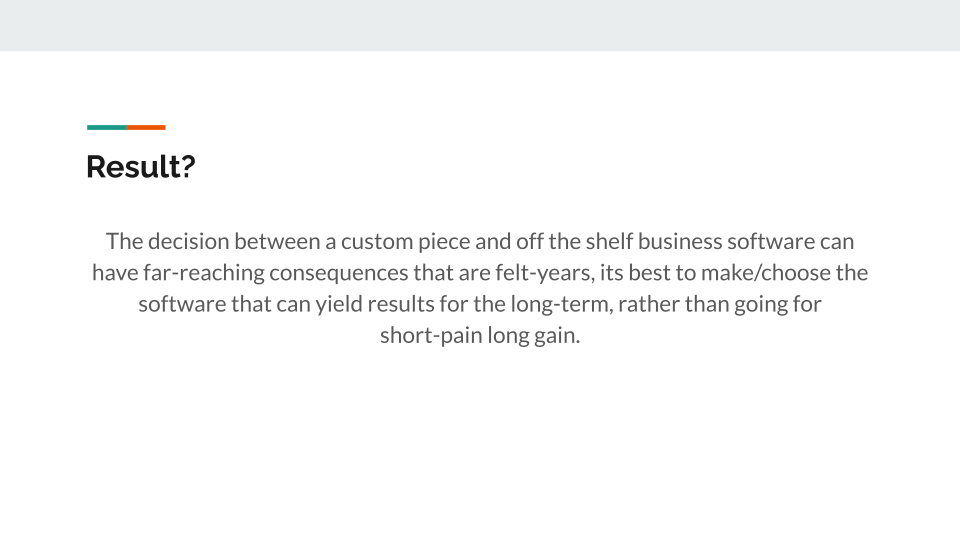


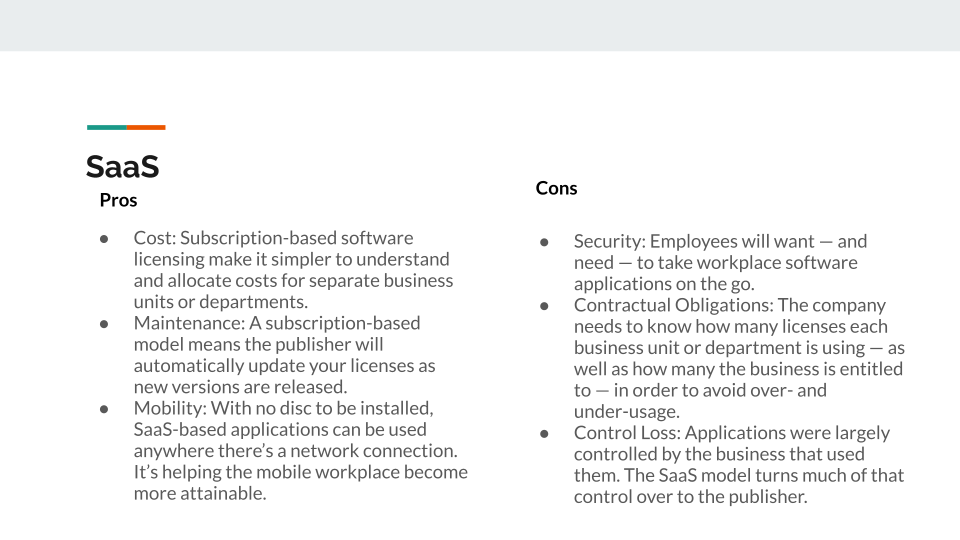


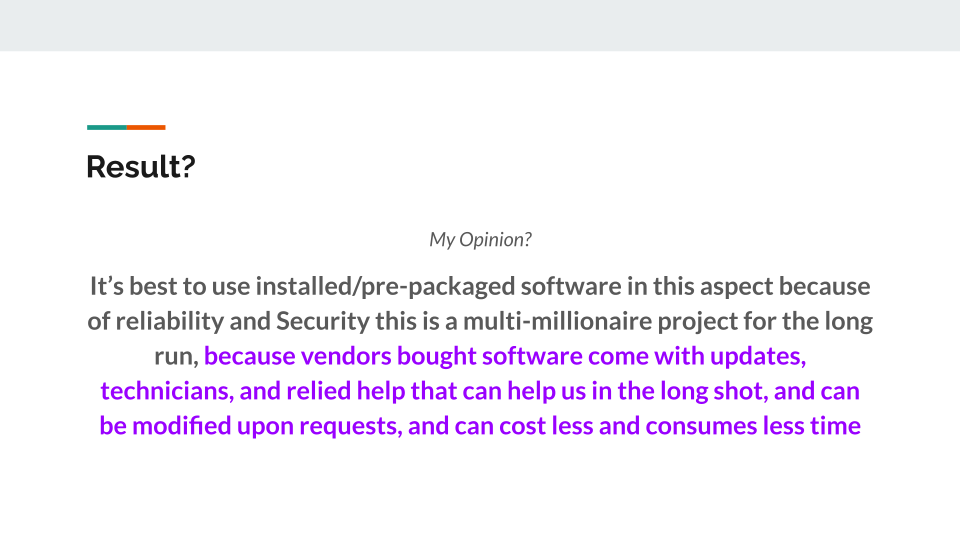


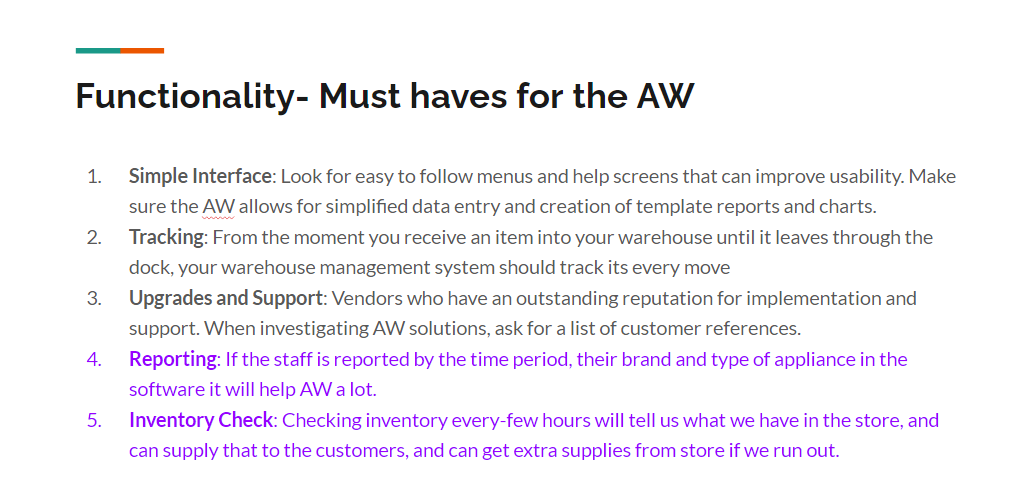


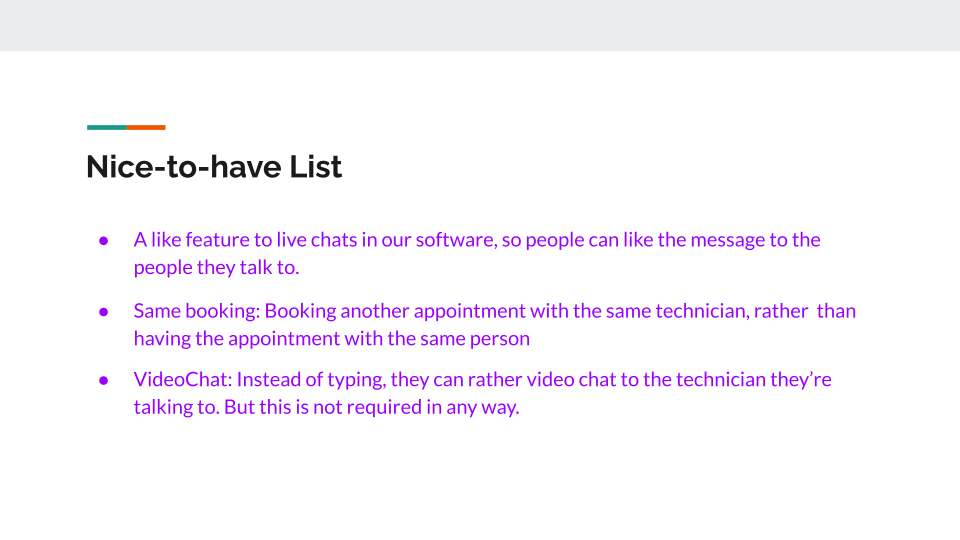




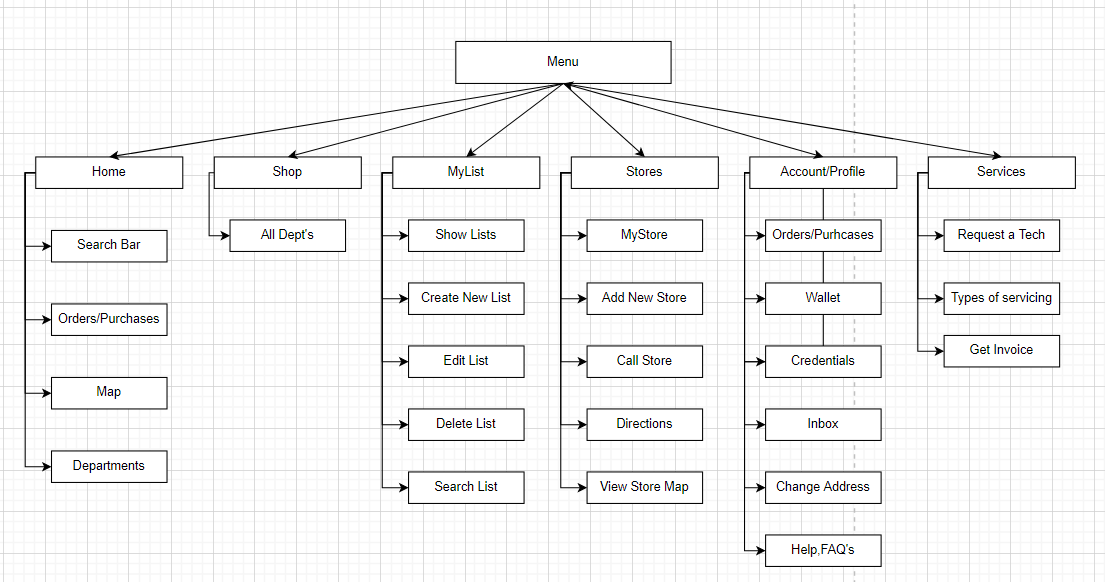








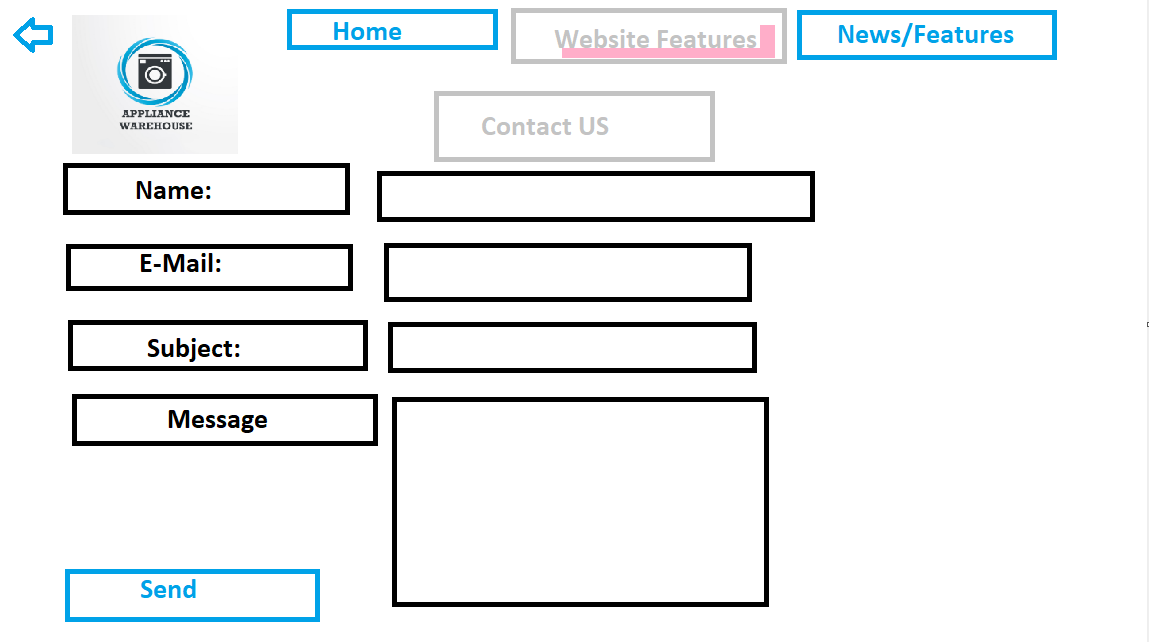
**Hierarchical Diagram**: Added invoicing, appliances, and the ability to edit, delete and search.



**Screen Examples**: Made prototypes based on AW.

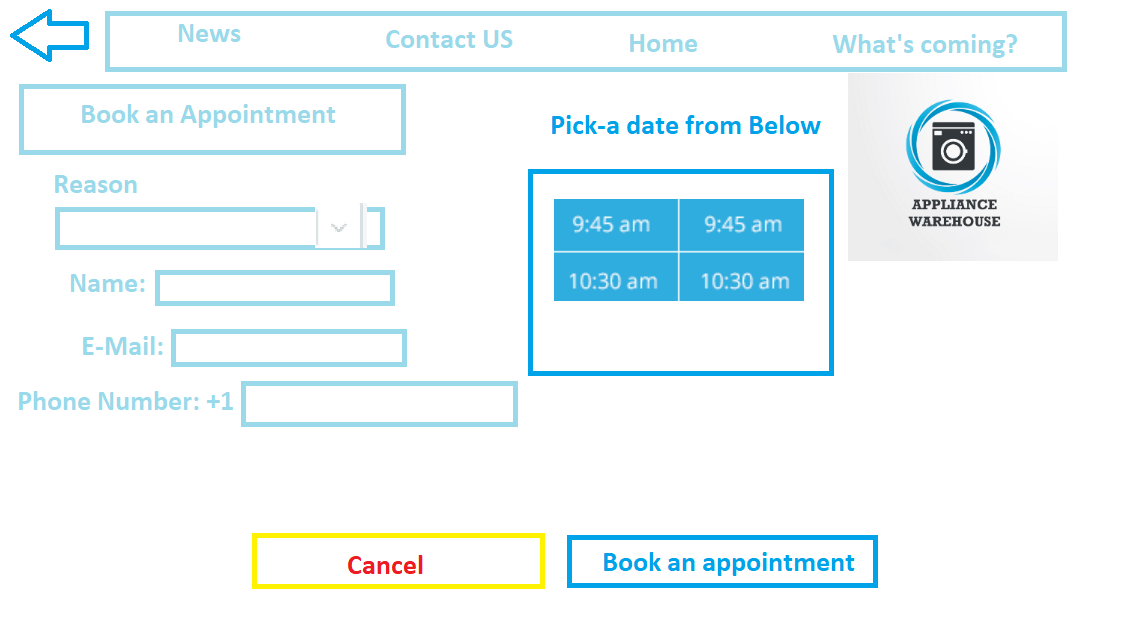
**Sample Contact US Page**:

**Input rules**: User inputs their name takes in String input, Email takes in string input which is unique, and the subject as well their message (string) and hits send.



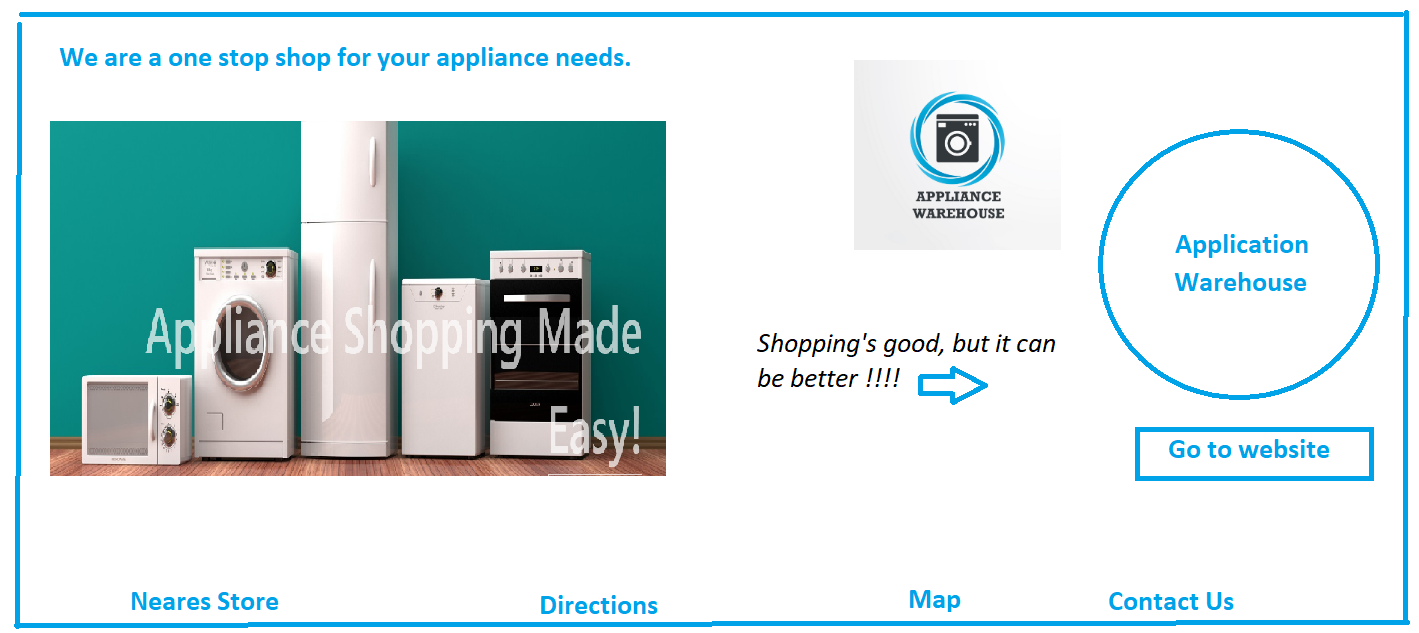
**Sample Make an Appointment Page**:

**Input rules**: User checks the reason for appointment choosing an option which can be selected from the dropdown, and enters the name **(string)**, email (**string)**, and the phone number which takes integers as input, as well as picking the date from below to book which uses **Date** feature., and **time** to choose from the box.



**Sample Opening Page**:

**Input rules**: User sees the website, and decides to click on go to website button and sees the home page.



**Prototyping**:

The new application can run on mobile platforms. Developers need to publish their apps on mobile platforms. Simply integrate a responsive web page to easily launch the app on a mobile platform, and resize the web page to fit the screen of your mobile phone, allowing users to see all or any perspective of your website or application. can do. clearly. This is the perfect developer to use responsive websites. This website can fit the website to the size of your mobile phone and will run smoothly on your mobile platform without major problems.

**Advantages of prototyping**:

1. Flexible design is easy to integrate new features.

2. Easy to find errors over the course of development.

3. We are able to find missing functionality easily.

4. Ideal for an online system as we can use the user feedback.

5. There is scope of refinement in the project .

6. Ensures greater level of customer satisfaction and trust and reliability.

7. It lets the users and the developer understand the project more easily and more deeply.

**Disadvantages of prototyping**:

1. Prototyping is expensive.

2. Requirements keep on changing over the development period with new changes made by the client.

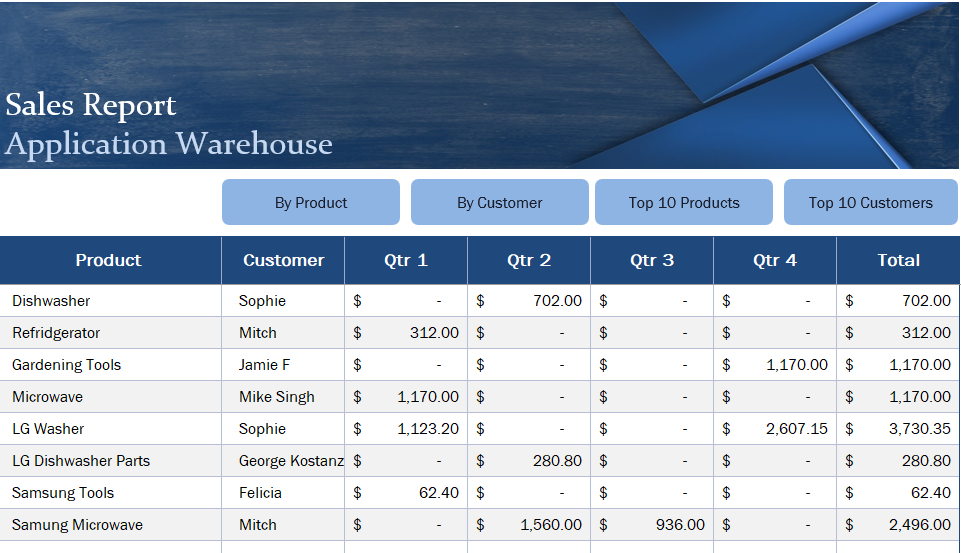
3. Over the course of development there are a large number of alterations in the project.

4. Customers might become anxious after seeing a prototype and demands for the product to be delivered to them very soon which is not possible.

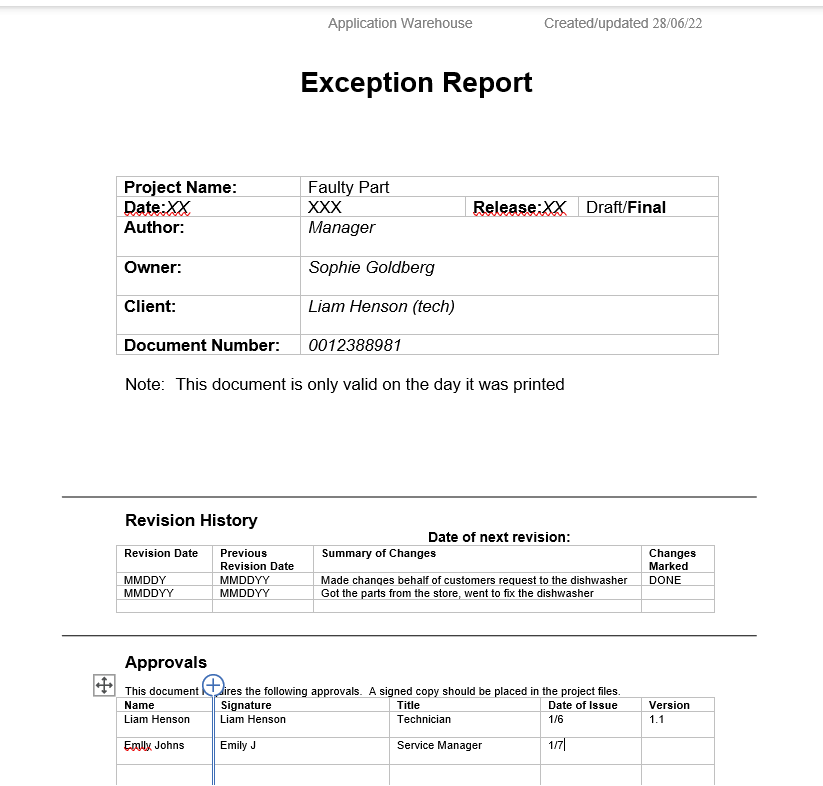
5. The prototyping system can be very complex and confusing for the user and the developer as well.

**Report Types**: Created mockup reports using the data from service records from AW.

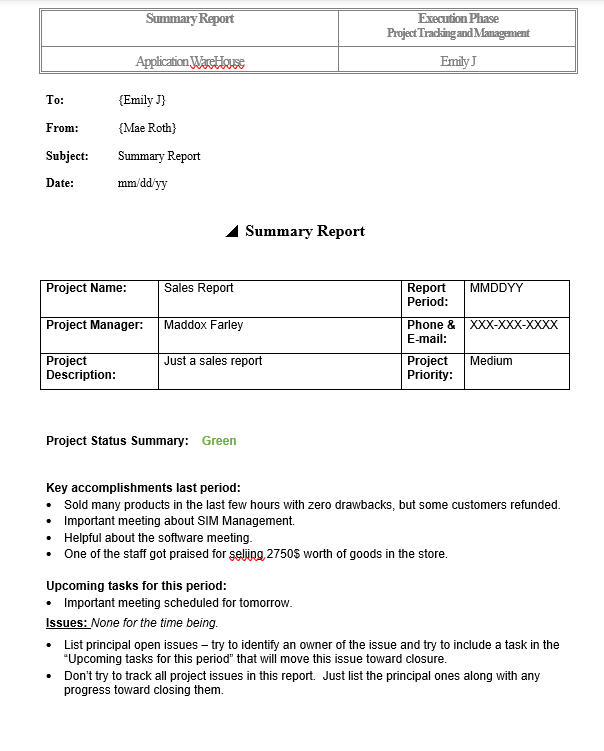
**Detailed Sales Report:**

****

**Exception Report:** Created a new exception report, and inserted data using records**.**

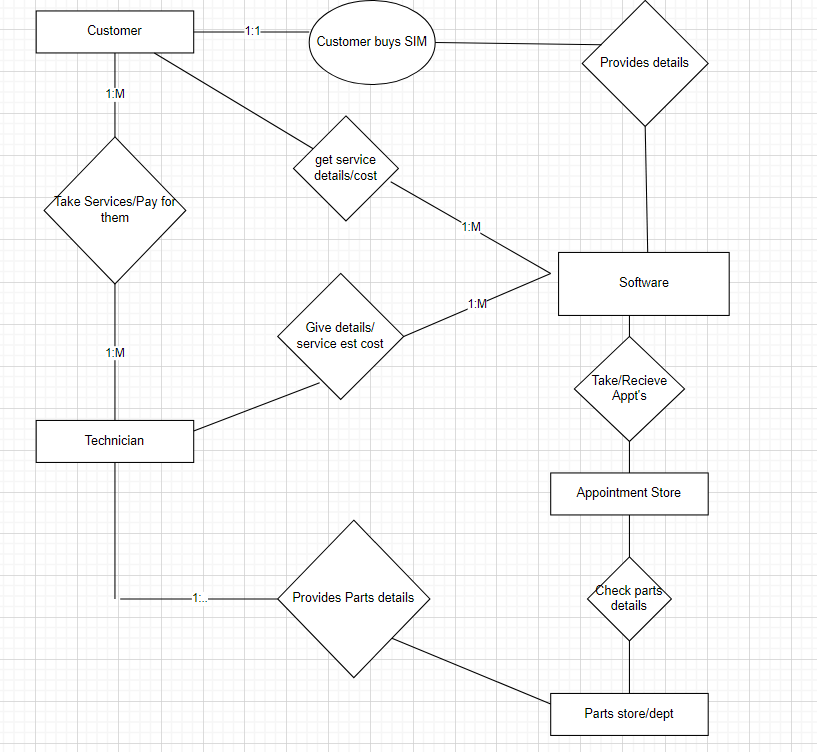
****

**Summary Report:** Made a mockup summary report from the emails

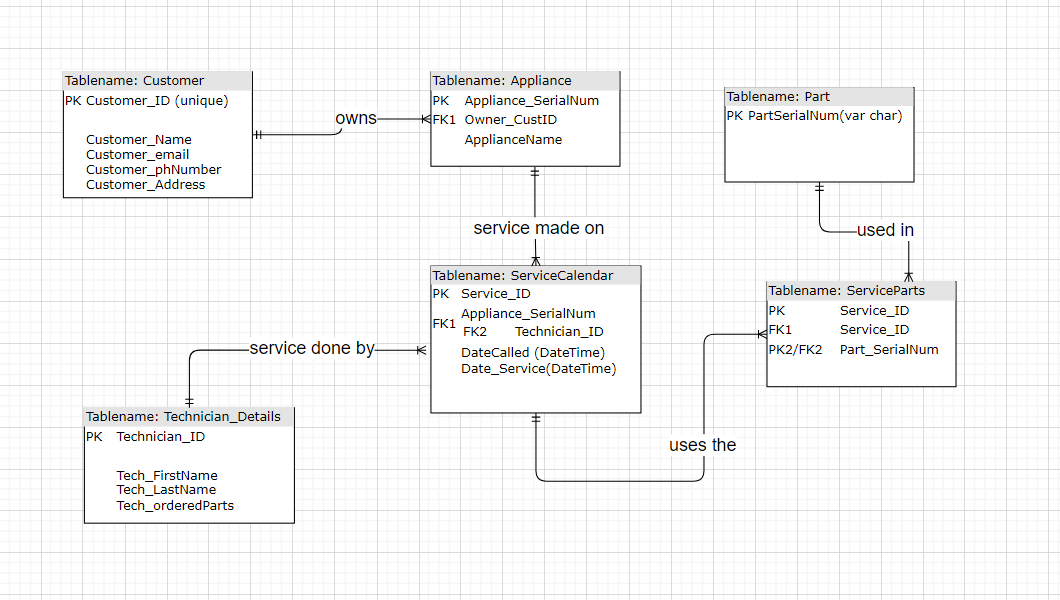
****

We can distribute most of the reports using E-Mail or any of the report generating softwares in the softwares we use, and set a command to generate commands on a basis and share with the workers and staff with their work generated E-Mails.

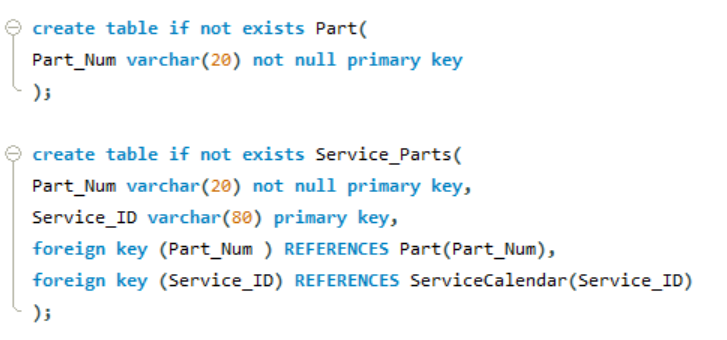
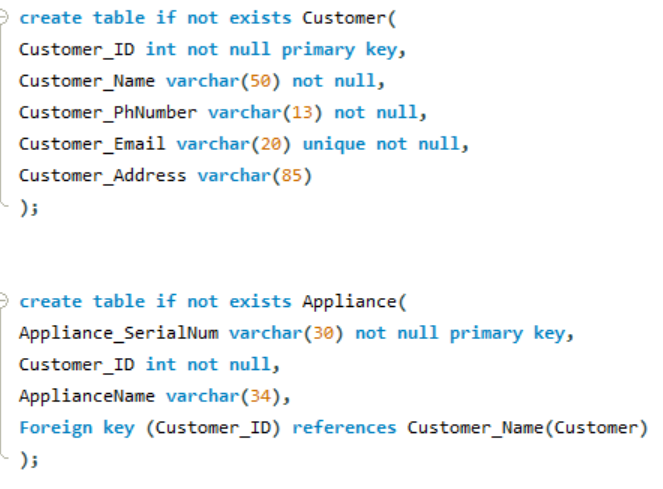
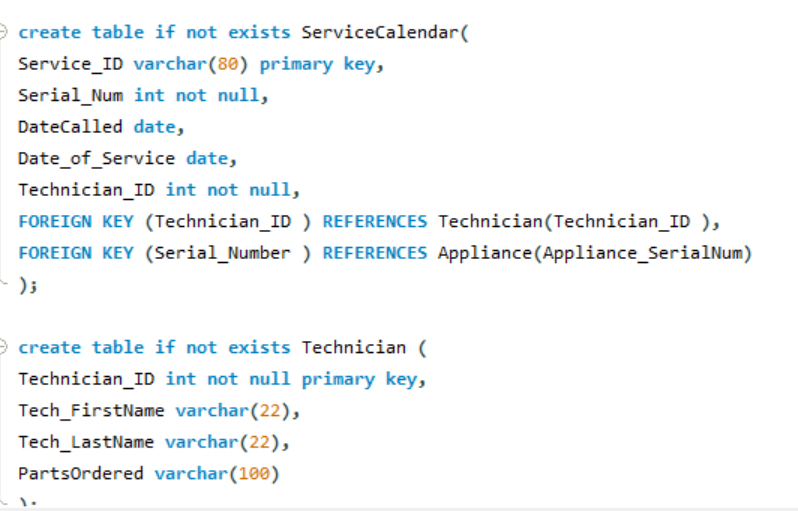
**ERD Diagram**: Made changes and added the 1:M relationship to the diagram, and removed SIM Entity.



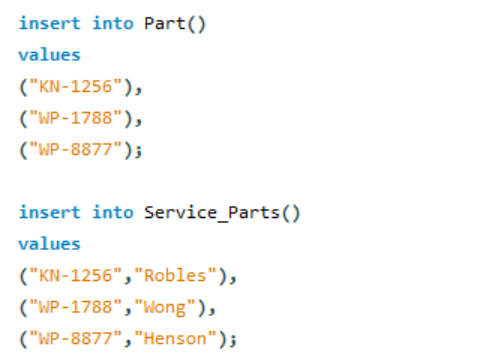
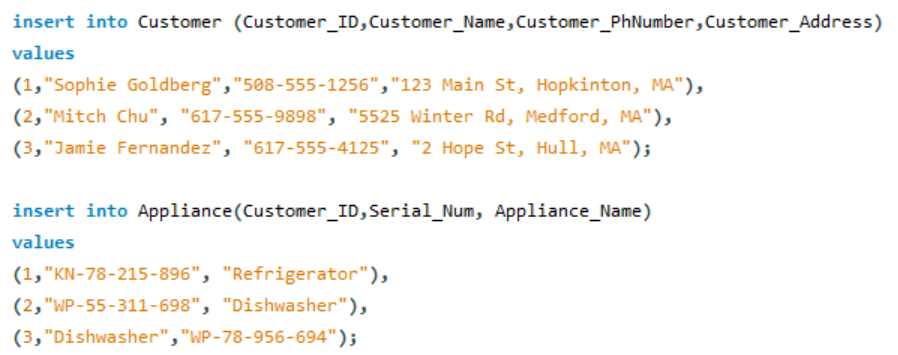
**Table Design**: Made changes and added customer’s new attributes, and added part name to the part table as well as added *appliance’s name* to appliance table.

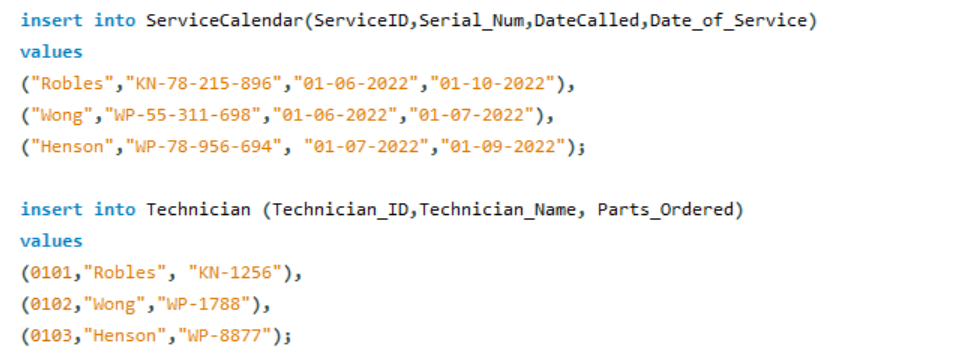


**Sample Data**: Table Creation : **Using the** **entities listed above**

**  
   
Data Insertion: 3 Dummy Data into each table**

**Tables: Customer, Appliance, Part, Service\_part, Service\_Calendar, Technician**





**ERP Softwares**: I found that Oracle Netsuite, MS Dynamic 365, and Oracle Cloud are the best ERP Softwares on the market.

| Oracle NetSuite  Customer Experience: Are less crowded, and mixed  Implementation: Less customizations   Less functionality  It is also a SAAS, and has medium cloud services  Flexibility, yes because it is a SAAS system, so common system.  Provides scalability via standardization since it wasnt personalized for some parts of a company  Organization is very small-medium size | MS Dynamic 365  Customer Experience: Very large buyers, resellers, and highly system integrators  Implementation: Complex, and hard to use UI  Functionality, I would say Medium, on a scale 6-7  Has less cloud, because they were converted to cloud now, before using GPN, and Axapta.  Cloud system so more customizations over Flexibility  Has a greater breadth of functionality to address more diverse business processes  Organization is med-large. | Oracle Cloud  Customer Experience: Full public cloud access to the user.  Implementation: More customization, and security offered.  High Functionality  Gives out unique security, and terms with regulations, and high firewalls with cloud environments.  Gives more flexibility b/w cloud and customers are Oracle Cloud  You can take advantage to your liking about the Oracle Cloud.  Large size corporation. |
| --- | --- | --- |

Based on the information above, the best I would think is the Oracle Cloud, because when it comes to security like Carli said Cloud would be the best fit, and you can customize the Cloud as we wish in Oracle Cloud, and also provides Scalability, Flexibility, and Security.

**B*elow are the images attached for deeper analysis in page 5***

**SIM Network**:

**Scalability**: We can add delivery system because more packages can be delivered by adding more delivery vehicles. More staff so more people can help if there are any customers if needed help, more servers so they can take traffic from online sources from the website. Daily maintenance to the server so they do not overheat from the server room.

**For System Requirements**: We can add testing tools, security, train the staff so they know how to handle the system, use a high-end security software with a decent pay, Oracle Cloud to store the data from the AW so that anyone from the AW can access it and make some changes if needed.

For the network, it would be best to go with star topology because they are fast and reliable since they only depend on single-hop features.

For processing I think it is best to go with online processing since everyone is mostly online, we can add more servers to take traffic and buy appliances from AW with no lags.  
**Security**: To protect the servers and machines, we will need….

1. Take inventory of what we have every evening before the store’s closing hours.
2. Gather the testing tools
3. Make sure everyone knows their role for the servers and the machines
4. Automated scans so we can rule out that every machine is there, and scan for any bugs in the servers
5. Sometimes automated doesn’t cover everything, so run manual analysis too.
6. Focus on main and vulnerabilities in the machines and servers so if we can get them fixed, its easy to focus on the smaller pieces.
7. Source code analysis by a technician
8. Lastly, run again and make sure everything runs smoothly.

**Comparisons for ERP Softwares: Deeper Analysis**

