ENSF 607 – Principles of Software Development
Fall 2023

Final Project:

Due Dates	
Post-Lab	**Submit before <u>23:59 on Friday , November 24, 2023,</u>**

<u>**The following rules apply to this lab and all other lab assignments in the future:**</u>

1. Before submitting your lab reports, take a moment to make sure that you are handing in all the material that is required. If you forget to hand something in, that is your fault; you can't use `I forgot' as an excuse to hand in parts of the assignment late.

2. <u>**20% marks**</u> will be deducted from the assignments handed in up to <u>**24 hours**</u> after each due date. It means if your mark is X out of Y, you will only gain 0.8 times X. There will be no credit for assignments turned in later than 24 hours after the due dates; they will be returned unmarked.

Project (100 marks)

Your group is to perform a software architecture assessment on the following use case.
The outcome is a PowerPoint presentation that should not contain more than 12 slides.
Important is that you apply the two laws of software architecture in the process. The evaluation is based on the WHY and your justification of it. Document your assumptions to support your case. **The following steps should be followed.**

Step 1: Determine the functional and non-functional requirements. Make assumptions if necessary**. (5 Marks)**

Step 2: Drive out which architectural characteristics would apply and why. (Justify). **(15 Marks)**

Step 3: Consider coupling. (Disintegrators/Integrators) to represent a coupling solution. **(15 Marks)**

Step 4: Consider contracts among services. What type (Strict or loose)? Justify and explain. **(15 Marks)**

Step 5: Present a diagram of your architecture. **(15 Marks)**

Step 6: Present a deployment strategy of your architecture that considers three environments (DEV, QA, PROD). **(15 Marks)**

Step 7: Discuss where you would apply fitness functions. What would you consider monitoring? **(15 Marks)**

Step 8: Present your findings in a PowerPoint presentation max 12 -13 slides. **(5 Marks)**

 We will have sessions for presenting your team's proposed solutions.

Case Description:

Best Appliances is a large home appliance chain that has numerous retail stores throughout Canada. When customers buy appliances and other electronic equipment, they can choose to purchase a support plan. Customer-facing appliance experts will then come to the customer's residence to fix problems with the device.

Things have not been good with the Best Appliances and if something isn't done soon, the company will be forced to abandon this very lucrative business line and fire all the employees including your architect team. You are tasked to come up with a new architecture to improve the reliability of the application.

1) The current trouble ticket system is a large monolithic application that was developed years ago.

2) Customers are complaining the appliance specialist is never showing up due to lost tickets.

3) Often the wrong appliance specialist shows up to fix something they know nothing about.

4) Customers have been complaining that the system is not always available for web-based problem ticket entry.

5) Change is difficult and risky in this large monolith – whenever a change is made, it takes too long and something else usually breaks (the product owner calls new feature deployments "Bug releases").

6) Due to reliability issues, the monolithic system frequently fails – they think it's' mostly due to an increase in usage and the number of customers using the system, but they're not sure.

The current process in the monolithic system:

1) Best Appliances repair experts are added and maintained in the system through an administrator, who enters their local, availability and skills.

2) Customers who have purchased the support plan can enter a problem ticket using the Best Appliance website. Customer registration for the support services is part of the system. The system bills the customer on an annual basis when their support period ends by charging their registered credit card.

3) Once a trouble ticket is entered in the system, the system then determines which appliance expert would be the best fit for the job based on skills, current location, service, area, and availability (free or currently on a job)

4) The Best Appliance repair expert is notified via a text message that they have a new ticket. Once this happens an email or SMS text message is sent to the customer (based on their profile preference) that the expert is on their way.

5) The Best Appliance repair exert uses a customer mobile application on their phone to access the ticketing system to retrieve the ticket information and location. The repair expert can also access a knowledge base through the mobile app to find out what things have been done in the past to fix the problem.

6) Once the repair expert fixes the problem, they mark the ticket as "complete", The repair expert can then add information about the problem and add the resolution to the knowledge base.

7) After the system received a notification that the ticket is complete, the system sends an email to the customer with a link to a survey which the customer then can reply to.

Current Architecture:
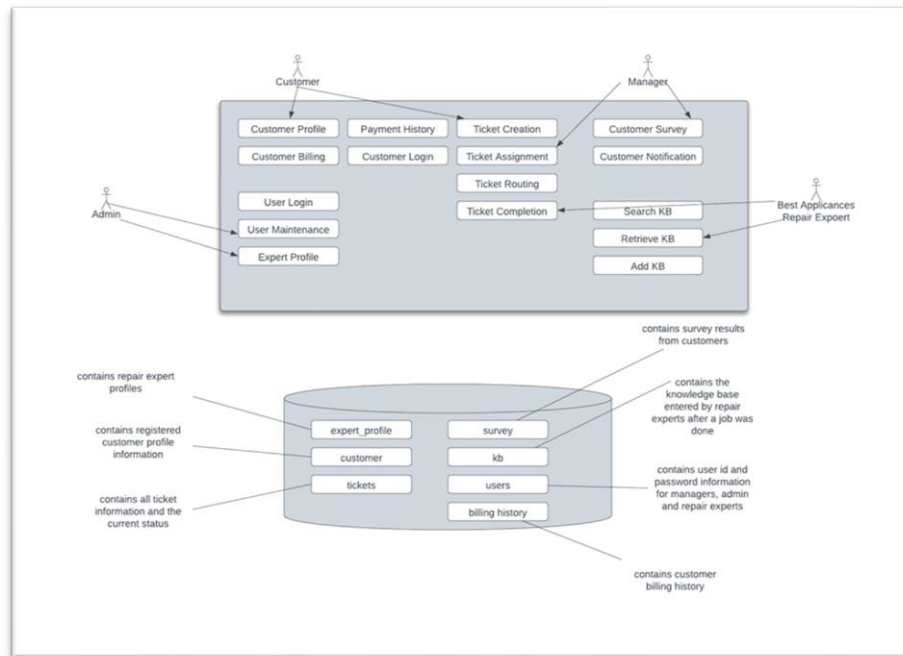
Ticket Workflow:
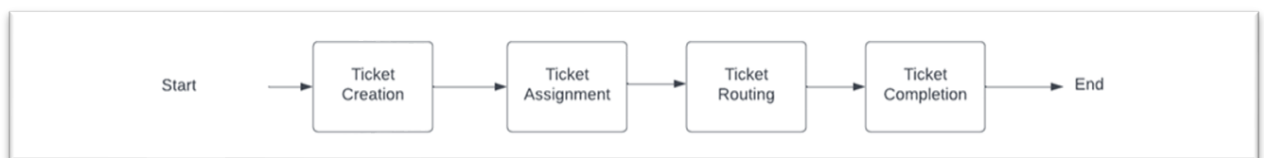
Additional Info:

1) Ticket assignment is a complicated process, and the algorithms change frequently to fine-tune the process.

2) Once a ticket is assigned to a repair expert, it is immediately routed to that expert. If it can't be routed, it is immediately reassigned.

3) Unit-of-work transactions are very important due to workflow issues with the current system.

4) Ticket creation is customer-facing and must be scalable and available for ticket entry.