Questrade Data Governance

- **What does Questrade do?**
  - Provides trading in stocks, cryptocurrencies, GICs, bonds, mutual funds, IPOs, precious metals, I could keep going. Questrade also provides a bunch of options for any client for whatever investment plan you are interested in. This could range from clients such as families, spouses to large companies and retirees. Plans can include:
    - Trust accounts, TFSAs, RRSP, RESP, RRSP, I could keep on going.
  - Questrade Trading and QuestMobile are products you made so that users can interact with their portfolios in an intuitive user-friendly interface.
    - Website offers news, real-time information and watchlists for stocks, ETFs, things you want to keep track of.
    - Mobile app where users can visualize real-time stock information, buy and sell ETFs and stocks.
  - (add a section relevant to the data governance portion)
- **Why work for this position/at Questrade?**
  - So my name is Kyle, and I'm a 3A mathematics student with interests in optimization, particularly variations of scheduling and the travelling salesman. I think stumbled upon this listing from Questrade as I want to align my work goals to work with something more quantitative. In school and some of my side projects, we sort of take it for granted much of the data we are given to solve problems, so it made me wonder why choose this particular dataset in general. I got to experience while data mining for ice cream truck images to train a computer vision model. How do we ensure that the data we are feeding into our models is accurate while still respecting the constraints we have on our resources? So when I saw the governance position on WW, I felt that it perfectly aligned with what I wanted.
- **Who is responsible for data governance**
  - It depends, but in general it should be the business side of the organisation and therefore not usually IT.
- **What is Data Governance?**
  - Data governance is the process of addressing data quality issues while providing a way for others to maintain a level of sanitation in our data so to speak with rules and guidelines.
  - It addresses data privacy and compliances as well, for instance, we would probably want to mask the SIN for users when we send data to the marketing department for example.
  - It also helps manage our data products, via metadata management.
  - *Discovery*
    - Understood all our data assets we have access to, whether that be on servers or in a cloud.
  - *Classification*
    - Assigning data to different categories, whether its customer data, product data, financial data. We can accomplish this using a wide variety of techniques, like classification models, but the key reason is to put a label on our data.
  - *Policy*
    - Documents that define how an organization use and manage its data, basically how should we utilize data processing and data management to ensure the data is accurate, consistent and accessible through an organization's systems.
  - *Rules*
    - This aspect is how we will enforce certain properties of our data, for example: if we have a social security number, we should mask it.
    - Data protection rules
      - Pertain to a specific type of data asset, for instance, credit card information should be masked etc.
    - Governance rules
      - A written description for those who use your data down the pipeline,
  - *Metadata (data descriptors)*

Questions

- What do you like the most about working at Questrade?
- How would you say progress is measured by? Like in software development, this could be measured in features, so I was wondering what that would look like for data governance.
- Would I go into the office everyday, is the position hybrid remote? If it is hybrid then how many days would I be required to go into the office.

Digital Innovation Services Student

- **What does FINTRAC do?**
  - **FINTRAC** (financial transactions and reports analysis centre of Canada) is the financial intelligence unit responsible for detecting, preventing, and deterring money laundering and financing terrorist activities while protecting the personal information under its control. Basically, it gathers intelligence and the info is used by other units to dictate their next decisions to make.
- **Why this position?**
  - I primarily chose this role since I wanted to build on my interests from my last work term working in web development, but I also noticed that in the role description, there would be opportunities to develop my knowledge on data science and machine learning, which has so far consisted of a combination of side projects and academic theory such as Monte Carlo algorithms and optimisation problems such as travelling salesman, scheduling, and knapsack. These problems have large implication in finance today, so although I know this is not a finance position, it still makes me wonder how the issues you solve here will impact models so relevant to the market and millions of Canadians everyday.
    - Monte Carlo simulation can be used to simulate risk analysis during a decision making process where random variables are employed.
  - Furthermore, I would like to use some of my coursework in Customer Experience, designing user personas mockups, etc. in an actual work environment, although I've seen them used
- `How to solve a problem`?
  - Following the principles of Test Driven Development, first and foremost, test test test! These blackbox test cases will ensure almost like a sanity check that your code works as expected.
  - When writing your code, there are some practises you should follow. You should ask yourself questions like why did you write X function, what is it supposed to do? The more simple the function is, the better.
  - So assuming you wrote your first pass of code, and you fail some test cases.
  - First of all, recreate the problem/bug, try to see if others can reproduce the bug, make it an issue on whatever repository you're using.
  - Using your favourite debugging method, I would try to locate the problem first using GDB, ur IDE's main debugger and step through the code to check where things start go wrong.
  - Now this is the fun part, and this is where you determine the scope of your problem to see if it was as simple as a syntax error or a large problem with a function. Typically the bigger the problem, the more I would subdivide the problem so I could tackle them individually via divide and conquer.
    - This process I described of asking yourself what happens when you return this or renders that is something known as rubbery ducky debugging, where you literally talk to it as if it were another developer almost, and walk through your thought process to describe what you are doing and why.
    - In 99% of cases, all bugs are fixed here. If not, it might be time to ask for another human's help!
  - Following your fix, make note of what you did wrong so others with similar issues can take note but also as a reminder to not make the same mistake!
- `What is an Event Driven Architecture`
  - In an ecosystem, each service takes action on an event it receives.
    - Ex. If you have an ecommerce system, when a purchase happens several things happen at once:
    - 1. email should be sent to user with receipt.
    - 2. decrease the counter in inventory for items bought
    - 3. fulfillment system should be notified
    - 4. And by existing, this includes the e-commerce system itself.

    - Event driven architecture works by setting up an intermediary event handler, so another service basically, that sends signals to an event subscriber, so one of our services that we described earlier.
        - This way, this promotes decoupling, this means that there is less dependency on services, so that if one thing goes down, it doesn't mean all other systems will go down.
        - Setting up an intermediary event router means that even if a particular system goes down such as the fulfillment system, this means that we can send another signal via the event router, and even send an error if handling for that is needed.
- `Decoupled`
    - It is the separation of front-end and back-end into separate entities so to speak.
        - We do this since in software development, if we make a change to one class, we don't want this modification to completely mess up the other classes it is dependent on.
- `Rest API (Representational State Transfer)`
    - Use standard HTTP methods such as `GET, PUT, POST, DELETE`.
    - Benefits include:
        - **Simple and standardized**, you don't have to worry about formatting your data
        - **Stateless/Scalable**: on the first part, stateless means that when you send/request data, the server treats it as a completely new request.
        - **High Performance**: they support caching.
    - In a web application, you probably want to implement CRUD functionalities, these are functions that communicate with the server via `create, read, update, and delete` and their rest API counterparts are `POST, GET, PUT, and DELETE`.
        - It would start with one of these, followed by an endpoint, website link, and have parameters you would parse in your request.
        - So it would go like `POST ENDPOINT PARAMS:toppings[pineapple, modzerella, bacon]`