**Project Plan**

1. **Motivation & Background Information**
   1. **Addressed Problem**

Hong Kong is one of the biggest player when it comes to supply chain, and it is not uncommon to hear news about losing the traceability of the goods. And most of the goods that need to be imported and exported require a declaration form that must be submitted. The number of those documents is continuously increasing and for a long time the approach taken was not a digitalized one yet. To address this problem, we need a solution capable of storing great amount of data and being able to be easily scalable as the data needed to be stored will only increase in the next few years.

* 1. **Current Approach**

It is unclear how exactly the Hong Kong government is storing the information about all these documents but in the past few years, we have seen an innovative approach by the government to keep the international competitiveness, by promoting the adoption of the electronic commerce.

**2. Problem statement, project objectives & scope**

* 1. **Problem Statement**

As the flow of information and data increases exponentially by time, we need a way to successfully store them in a safe way and if needed, scale the whole system without making the service unavailable. Not only the solution should be able to scale easily but also provide great performance and responsiveness and ultimately being extremely reliable.

* 1. **Goal Of The Project**

Design a reliable distributed database aimed to store a great number of documents, not only capable of handling multiple tasks at the same time but also fault tolerant and scalable as the number of stored information increases. After the designing part, the focus will be on implementing and developing a prototype which will be tested to check its performance.

* 1. **Scope Of The Project**

This project will include the following items:

1. Design of the distributed database architecture
2. Development of the prototype
3. Performance Testing

The precise schema of the import and export documents will not be used. The main constraint is that although the prototype will be a distributed databases, there are not sufficient resources to test it on different nodes around the world and the testing will be conducted using multiple virtual machines set as nodes.

**3. Major technical components**

**3.1 Technical Components**

The end goal of the project is developing a prototype of distributed database and in order to achieve this goal, there are multiple requirements to take into consideration.

1. The first goal is to decide which approach to take when designing the distributed database, from data replication or data partitioning.
2. Choose a Consensus Algorithm which fit the designed database
3. Define protocols to manage distributed transactions
4. Choose the optimal query processing algorithm for the project use case.
5. Define system security protocols and access control flow.

**3.2 Breakdown technical components**

1. When discussing which approach to take when building a distributed database, the decision of choosing data replication or data partitioning is crucial as it will affect the decision of other protocols as well. Each of these two approach has its own benefits and cons and an initial research on the adaptation in our use case will decide which one to implement.
2. A Consensus Algorithm is very important as it will be responsible to ensure the consistency of the data among all the nodes in the system. It allows different nodes to reach a “consensus” state when it comes to accept new data and distribute it across all the participant of the system. There are different available algorithms such as Paxos and Raft, and all of them have their own trade-offs when it comes to performance, fault-tolerance and complexity
3. As any other database, we need to ensure the ACID properties as well, and this is the reason why there is the need of algorithms and protocols that can guarantee the data can obtain all those properties.
4. Having performance as one of main reasons that lead the use of distributed database over the traditional relational database, the selection of the correct query processing algorithm is essential. We could implement multiple techniques such as query decomposition, query routing, and parallel execution.
5. Security measurements take into consideration a lot of aspects such as data encryption algorithm, secure communication between the different nodes in the system, authentication, system backup and many others. A lot of initial research will be focus on these aspects as some of them are quite challenging to tests properly.

**4. Expected results & deliverables**

**4.1 Expected tests cases**

The test cases conducted for the project would probably consists in utilizing multiple VMs as nodes or also multiple containers with each an instance of the node to replicate the behaviours of a distributed database. Different test will be conducted to analyse different aspects of the database such performance testing, functional testing, fault tolerance testing .

**4.2 Deliverables**

1. Design Report: A comprehensive report about the design of the distributed database system intended to address the identified issues.
2. Distributed Database System: A fully functioning distributed database system. The system will be developed using appropriate programming languages and technologies (after deciding which one would be the most appropriate), and should be ready for implementation and testing.
3. Performance Testing Plan and Report: A comprehensive plan for testing the system's performance, scalability, and reliability, as well as a report summarizing the results of these tests.

**5. Project schedule**

**5.1 Tentative Timeline**

September 2023 – October 2023 : Analysis of existing solutions and defining the cons and pro of specific approach over the others. By the end of October, chose between data replication or data partitioning.

November 2023: Research about all the protocols needed to develop the database and choose the one that are going to be used for the prototype.

December 2023: Define the security measurements to be taken and design the database architecture.

January 2024 – March 2024: Database prototyping. The final result should be a functional database.

March 2024 – April 2024: Testing and eventual bug fixing and adjustments.

A screenshot of a calendar

Description automatically generated