**Andrey Cherdakov**

(keygard100@gmail.com)

**Test task**

**“XML Reader”**

Israel 2021

Content

Introduction ………………………………………………………………….2

Technical Task ……………………………………………………………….3

Algorithm ……………………………………………………………………4

UML Diagram ……………………………………………………………….5

Database structure …………………………………………………………...6

POM.xml……………………………………………………………………..7

**1. Introduction**

The software is designed based on the technical specifications issued by the Customer, which is also given below.

The developed system consists of one part:

- Server receiving requests and processing them, loading XML file, processing and filling the database

**2. Technical task**

The system must allow the following:

* Reading request files from file system folder every one hour
* Duration and folder location can be changed
* On memory database to hold the fresh data

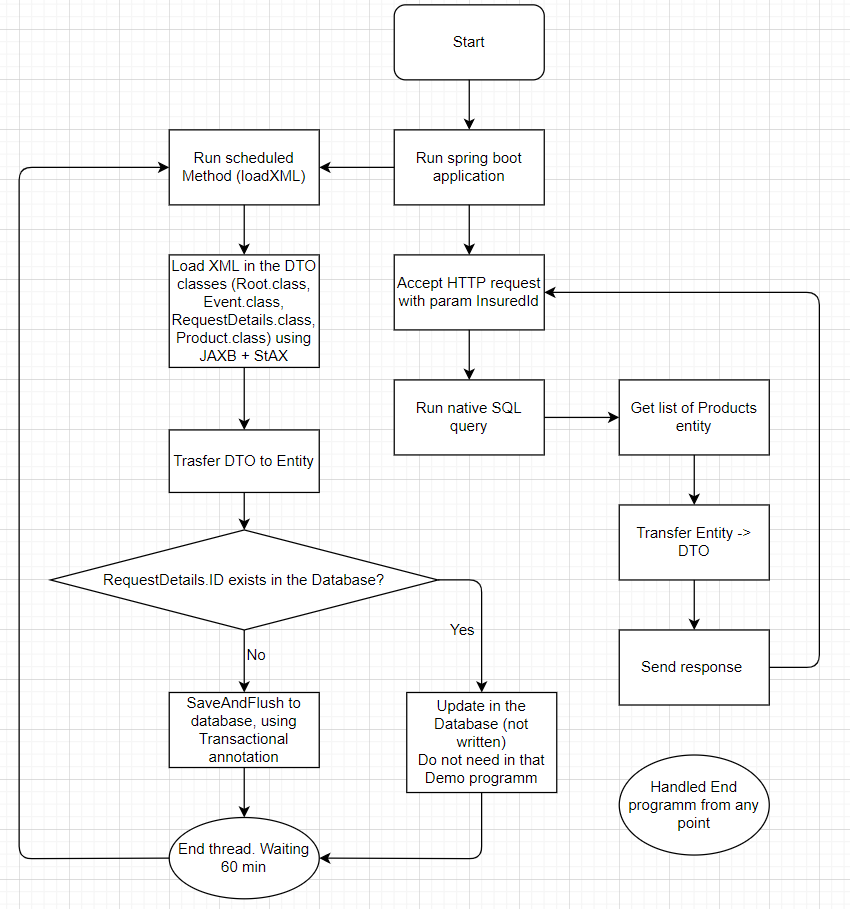
You need to provide:

* Rest API to present products by InsuredId as following:
  + Result must be in JSON format
  + Result should include all products related to same InsuredId grouped by SourceCompany
* A scheduled process to read data each hour from folder

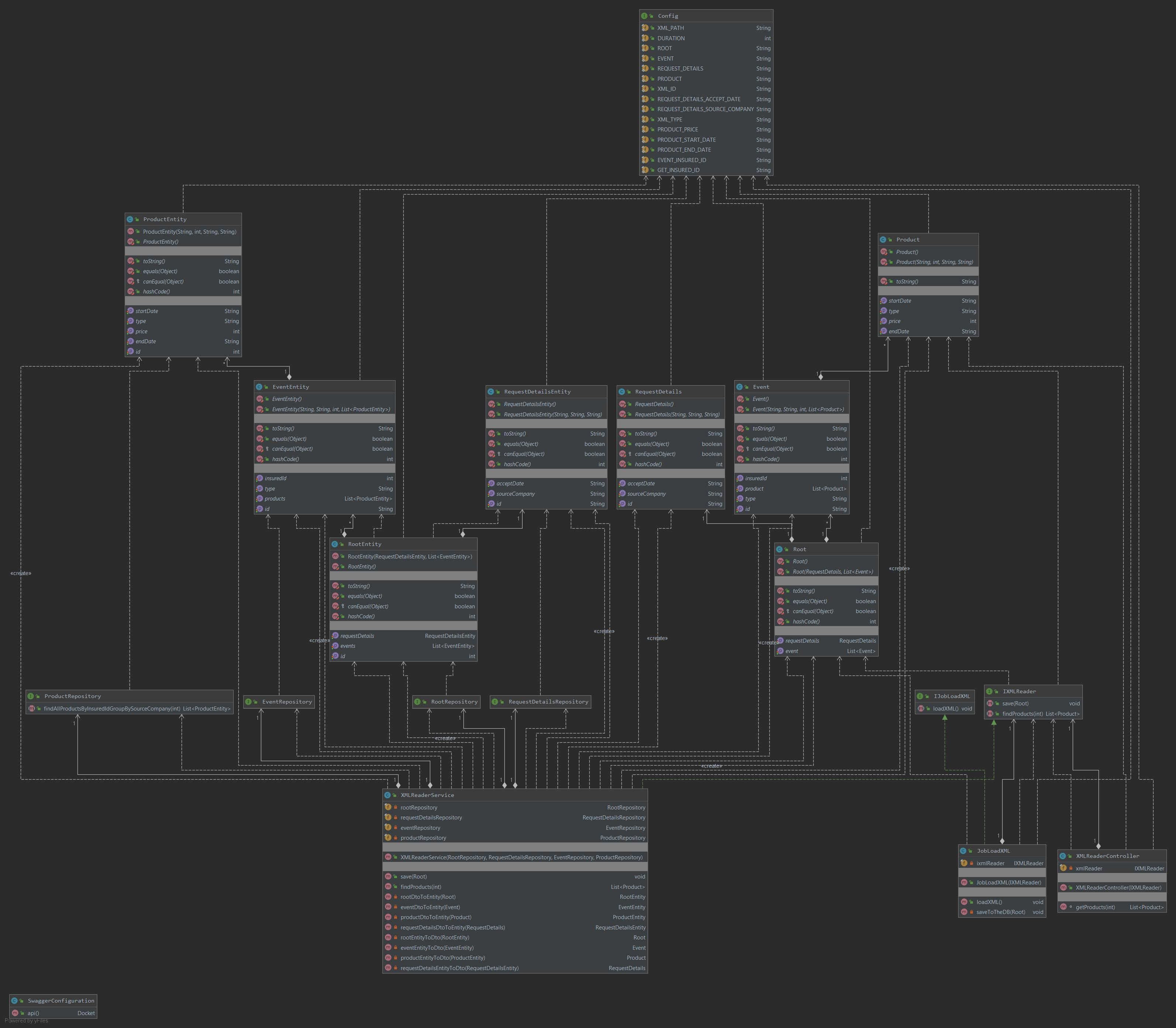
Technical requirements and guidelines:

* Backend : Spring Boot
* Submission must be a day before the interview.
* Submission can be in github repository.
* If you have any major concerns feel free to reach out

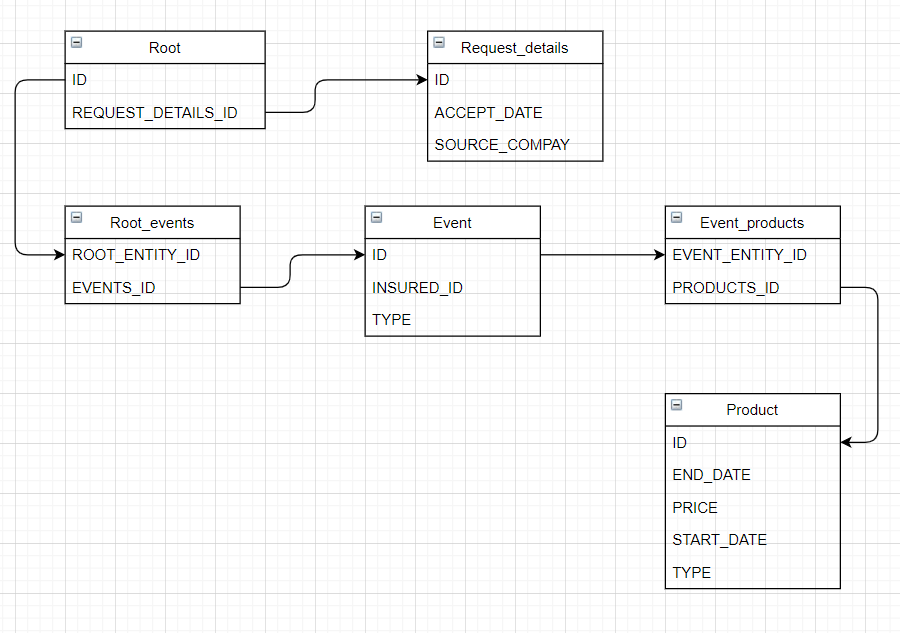
**3. Algorithm**



**4. UML**

****

**5. Database structure**



**6. POM.xml**

<?xml version="1.0" encoding="UTF-8"?><project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"> <modelVersion>4.0.0</modelVersion> <parent> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-parent</artifactId> <version>2.5.3</version> </parent> <groupId>org.example</groupId> <artifactId>XMLReader</artifactId> <version>1.0-SNAPSHOT</version> <dependencies> <dependency> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-web</artifactId> </dependency> <dependency> <groupId>com.h2database</groupId> <artifactId>h2</artifactId> <scope>runtime</scope> </dependency> <dependency> <groupId>org.projectlombok</groupId> <artifactId>lombok</artifactId> <optional>true</optional> </dependency> <dependency> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-test</artifactId> <scope>test</scope> </dependency> <dependency> <groupId>javax.xml.bind</groupId> <artifactId>jaxb-api</artifactId> </dependency> <dependency> <groupId>javax.activation</groupId> <artifactId>activation</artifactId> </dependency> <dependency> <groupId>org.glassfish.jaxb</groupId> <artifactId>jaxb-runtime</artifactId> </dependency> <dependency> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-starter-data-jpa</artifactId> </dependency> <dependency> <groupId>io.springfox</groupId> <artifactId>springfox-swagger2</artifactId> <version>2.9.2</version> </dependency> <dependency> <groupId>io.springfox</groupId> <artifactId>springfox-swagger-ui</artifactId> <version>2.9.2</version> </dependency> </dependencies> <build> <plugins> <plugin> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-maven-plugin</artifactId> <configuration> <excludes> <exclude> <groupId>org.projectlombok</groupId> <artifactId>lombok</artifactId> </exclude> </excludes> </configuration> </plugin> </plugins> </build></project>