**Project Title**

Pharmacy Solution Provider

**Authors**

Luai Iwidat

Mohannad Jamal

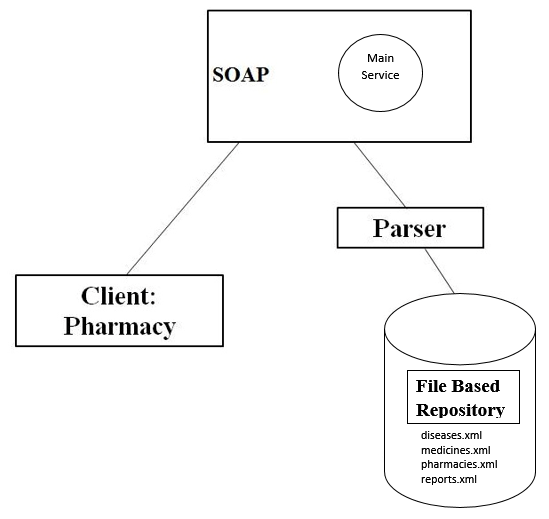
**Abstract**

Pharmacy Solution Provider is a web service that provides useful info which can be heavily used by pharmacies. It gives the ability to clients to look up what are the diseases, their symptoms and cures, the preferred medicine and where to get it from several pharmacies.

**Proposed Architecture**

The system will consist of **one** *service provider.* The provider will host services related to diseases, medicines and pharmacies. The 3 classes (which are obtained after parsing xml files) are used by different services interchangeably to provide a specific goal.

The following figure provides a simple architecture to our project:

****

2 Parsers (DOM):

1. Disease, medicine and pharmacy parser.
2. Writing parser.

**Client**

The client will be a GUI desktop application representing a pharmacy that will use the services to get valuable info, diagnose diseases and order medicines (with report keeping track).

**Service Providers**

* **Pharmacy Solution Service:** (Pharmacy Solution Service.java)

It features 7 services:

1. getDisease(int diesaseId): Gets the disease with specified id.
2. getMedicine(int medicineId): Gets the medicine with specified id.
3. getPharmacy(int pharmacyId): Gets the pharmacy with specified id.
4. diagnose(String[] symptoms): Gets a list of possible diseases for given symptoms in order to “diagnose” the sick. **Note that this service didn’t work as many unknown bugs occurred.**
5. getCure(int diseaseId): Gets a list of medicines for the specified disease.
6. getPharmacies(int medicineId): Gets a list of pharmacies that are selling the specified medicine.
7. Order(int pharmId, int medId): Adds a report of the chosen pharmacy to get a medicine from to reports.xml.

**Enabling Technologies**

**Client**

Java

**Service provider**

SOAP

Java

Glassfish

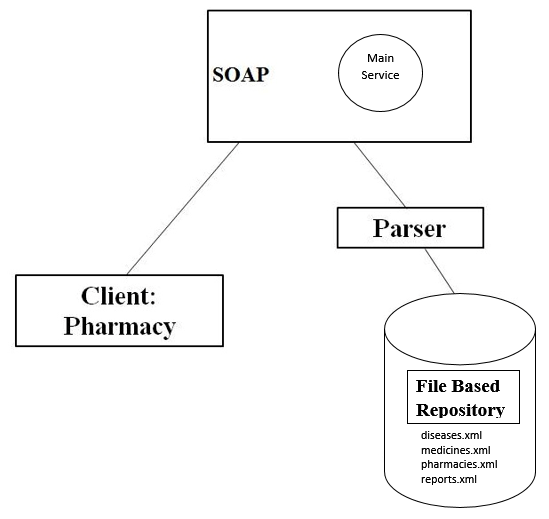
File Based Repository (XML files)

**Future work**

Our inspiration for our project was building a suitable system for pharmacies in Palestine and connecting them together, with a database (not a real database in our project) containing valuable info about diseases and medicines. Our project could be tweaked and fixed so that it can be suitable for larger purposes, and benefits all pharmacies in our country.

One way of making our project better is using web service composition. For example, instead of manipulating disease, medicine and pharmacy data under one service, we could make a service for each one. The services then can invoke their selves and others to achieve different goals in a more flexible way. The following picture shows web service composition architecture:

SOAP

****

2 Parsers (DOM):

Disease, medicine and pharmacy parser.

Writing parser.