“INTERDRUGS” SEMANTIC WEB APPLICATION SYSTEM DESIGN REPORT

1 INTRODUCTION

1.1 Purpose and Scope

Multiple diseases may happen to a single person. To provide holistic medical treatment, healthcare providers always need to prescribe a list of drugs, targeting at different diseases at the same time. However, if any one of the drugs interacted with another, the prescription might bring a negative result that neither the doctor nor the patient wants to see. Meanwhile, it is often mentally heavy burden to memorize relationships among diseases, drugs, and interactions. The result is, a slip in the doctor’s mind could cause serious medical accidents.

To help fill this gap, we developed a semantic web application named “InterDrugs”. The InterDrugs system does not entirely replace doctors’ work, but plays a supporting role in the process of healthcare providers’ decision making.

1.2 Project Executive Summary

We built a semantic web application that conveys skills learnt from Semantic Web course, and could be applied in medical field. Source codes and ontologies are published open-access. A package of materials involved could be reached from this GitHub repository given as below:

https://github.com/PietroFerretti/FinalProject

1.3 Document Organization

This report consists of 5 parts: introduction, system architecture, database design, user definition and interface design. Graphs and tables are applied to explain concepts, structures and screenshots.

1.4 Points of Contact

This project is carried out by Group ◆[1] at ◆ University. Total two members of the group undertook responsibilities of a system designer, database constructor, product manager, information architect and user experience designer. Here are the contacts of project members:

(Real member information omitted…)

1.5 Project References

[1] D2R Server publishing the Diseasome Dataset [EB/OL]. http://wifo5-03.informatik.uni-mannheim.de/diseasome/. 2015-10-19.
[2] D2R Server publishing the DrugBank Database [EB/OL]. http:// wifo5-03.informatik.uni-mannheim.de/drugbank/. 2015-10-19.
[3] Wishart, David S., et al. "DrugBank: a comprehensive resource for in silico drug discovery and exploration." Nucleic acids research 34.suppl 1 (2006): D668-D672.
[4] Mulder, Steve, and Ziv Yaar. The user is always right: A practical guide to creating and using personas for the web [M]. New Riders, 2006.

1.6 Glossary

- ***Disease***: 4,300 disorders and disease recorded as RDF data on D2R Server, as well as the genes causing some of them. These diseases are classified by categories.
- ***Drug***: detailed drug (i.e. chemical, pharmacological and pharmaceutical) data with comprehensive drug target (i.e. sequence, structure, and pathway) information.
- ***Diseases under control***: diseases that have at least 4 possible drugs. With a recognition of a disease "under control", an involved patient user of our

[1] We replaced group number and university name with "◆" symbols, and omitted member names/contacts, to support ***blind review*** regulations. However, a "points of contact" section is necessary in a professional development report.

application could be comforted with the scientific condition towards his/her health guarantee.

- ***Diseases under investigation***: diseases that have at most 2 possible drugs. With a recognition of a disease "under investigation", an involved doctor user of our application could be reminded being cautious in deciding treatment programs.
- ***Treatable disease***: diseases that have at least 1 possible drug.
- ***Interacting drugs***: drugs that are interacting with the current drug.
- ***Unsafe drugs***: illicit drugs and withdrawn drugs.

2 SYSTEM ARCHITECTURE

2.1 System Overview

With a user's input of disease names, the system retrieves in the ontology we built, which combined third-party data sources, and then outputs drugs that are possible to cure these diseases, as well as giving interacting drugs a page highlighting effect, reminding healthcare providers to avoid drug conflicts.

Figure 1 maps the process of "InterDrugs" application.

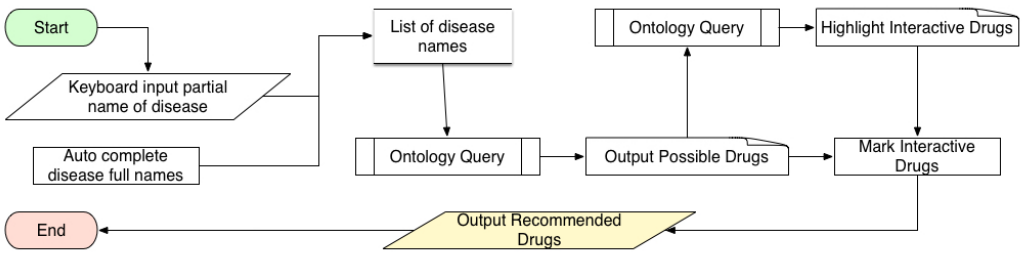

Fig 1. Flowchart Of "InterDrugs" Application Process

2.2 Software Architecture

The "InterDrugs" application is supported by several files and services, the data communications among which are hidden against users. The user interface is a webpage in HTML format. Figure 2 depicts the software architecture of "InterDrugs" application.

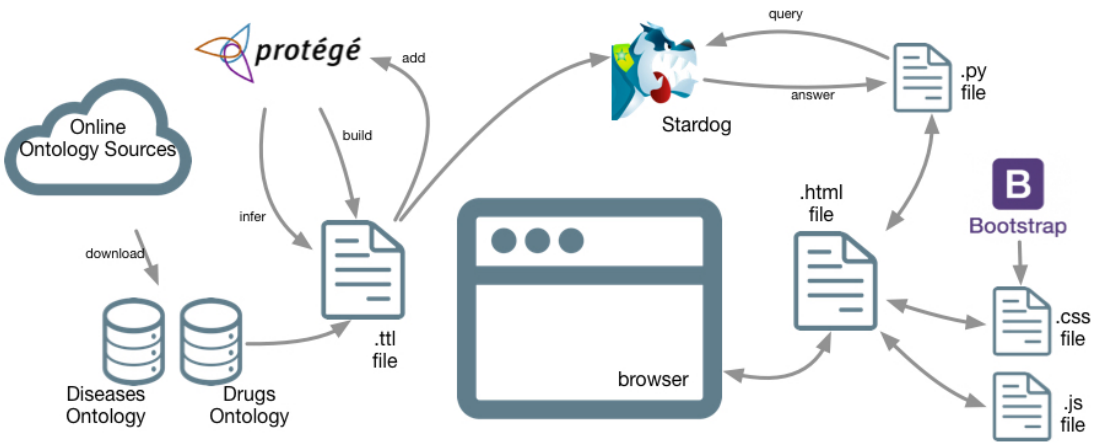

Fig 2. Software Architecture of "InterDrugs" Application

An HTML file organizes the function components shown to a user. A CSS file formats the page and optimizes visual design. A JavaScript file catches user actions, while input data is passed to a Python program that generates a SPARQL query. This SPARQL query is sent to Stardog database manager, which retrieves our ontology, and then returns wanted results to Python program. Next, results is arranged in human-readable format by JavaScript and illustrated on the browser by CSS and HTML.

The ontology, as uploaded beforehand to Stardog, was created with Protégé originally, in Turtle format. Triples and instances were downloaded from the Internet and added into the Turtle file. Based on axioms and definitions on classes and properties, the Pellet reasoner of Protégé produced inferences, which enables the "InterDrugs" application to provide reasoned results.

2.3 Design Constraints

We had great projects for our application, but due to time constraints and more importantly because of our lack of skill and experience in web programming (neither of us ever actually learnt JavaScript or CSS), the application is far from completed. This is what is missing and we would have implemented if we had more time:

- A page to search for genes and diseases related by the same genes. In the code we left a draft (annotation formatted), but it's not working yet;
- A filter for unsafe drugs;
- Some graphic design and a better formatting;
- A description for every interaction (not in our ontology so we would have used an external SPARQL endpoint dynamically);
- And more.

We're still learning but we did our best.

3 DATABASE DESIGN

3.1 Data Source Selection

The InterDrugs application was developed with Data-Oriented Design (DOD) methodology. In other words, the application has not been sketched out until valid data sources were confirmed. Dominantly, the data sources imported in our application were examined according to requirements of system quality indicators, which are essential for us to provide a robust application.

Four aspects of system quality were taken into consideration: usability, affordability, interchangeability and sustainability. Table 1 exhibits correspondence relationships between system quality and data source quality in our application.

Tab 1. Correspondence Relationships between System Quality and Data Source Quality

System Quality Indicators	Data Source Quality Indicators
Usability: Effectiveness	Accuracy Completeness Relevance Reliability
Usability: Simplicity	Appropriate Presentation
Affordability	Accessibility
Interchangeability	Consistency across Data Sources
Sustainability	Update Status

3.2 Data Structure

The ontology of "InterDrugs" application consists of three types of classes: Diseases, Drugs and Genes. Figure 3 shows a hierarchy of the classes.

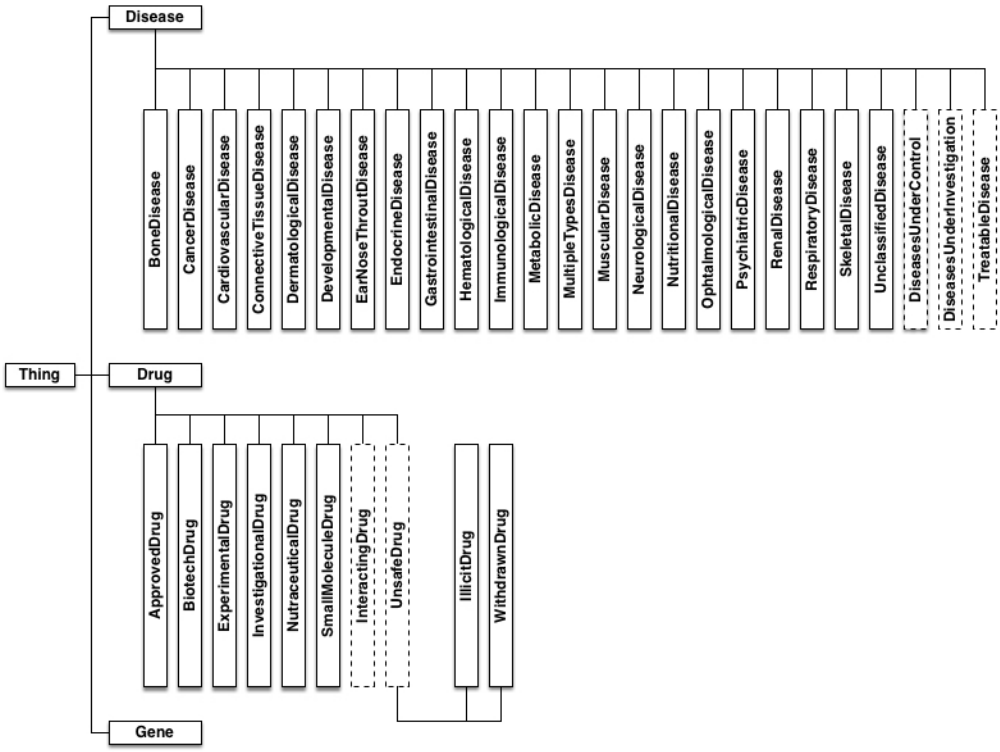

Fig 3. Ontology Structure of "InterDrugs" Application

Diseases are classified into 22 classes, i.e. "BoneDisease", "CancerDisease" etc., as defined by the online data source "Diseasome". Some diseases are recorded caused by certain genes, which then mutually become "related diseases". This relationship enables reporting related diseases by the application, which is yet not realized in current version. The 22 classes are predefined, instances of which are downloaded and imported by class labels.

The other three classes of disease are defined with class restrictions, i.e. "DiseaseUnderControl", "DiseaseUnderInvestigation" and "TreatableDisease". Instances of these classes are inferred results.

Similar structure is the class of "Drug" and its subclasses. "Gene" class is affiliated to "Disease" class.

3.3 Data Reuse

Two external ontologies are reused in this project: *Diseasome* and *DrugBank*. The two ontologies are both created by Research Group Data and Web Science at the University of Mannheim, published with D2RQ Platform and can be reached via

SPARQL endpoints. Links to the two data resources are given in section 1.5 of this report.

- From *Diseasome* we retrieved the diseases, classified by category, a list of the drugs that can treat each disease (the connection between *Diseasome* and *Drugbank*), a list of the genes and the diseases that they can affect.
- From *Drugbank* we used their list of drugs, their types and the existing brand names for every drug.

Our ontology is open to public from the following URLs:

- Ontology (without inferences): http://huiqingao.com/interactingDrugs.ttl
- Ontology (with inferences): http://huiqingao.com/interactingDrugs-inferences.ttl

3.4 Ontology Inferences

Our ontology uses inferencing for further classification of drugs and diseases, to make searches faster and easier (for example using filters) and building more structure.

Furthermore inferencing is used to make possible searching for diseases related by the same genes, or related drugs that can treat the same family of diseases.

3.5 Query Examples

3.5.1 Searching for a disease

Figure 4 is a screenshot of disease searching query. We can use this query to find a disease in our database starting from an input by the user, in this case "diabetes".

```
PREFIX : <http://huiqingao.com/ontology/interactingDrugs.ttl#>

SELECT ?disease ?label WHERE {

  ?disease a :TreatableDisease;
           rdfs:label ?label.

  FILTER regex(?label, "diabetes", "i" )

}
```

Fig 4. Query Example: Searching for A Disease

"TreatableDisease" is defined as a disease for which our database knows a feasible drug.

3.5.2 Finding possible drugs

After finding the correct result we can look for possible drugs to treat that specific disease. Figure 5 is a screenshot of the query.

```
PREFIX : <http://huiqingao.com/ontology/interactingDrugs.ttl#>

SELECT ?drug ?label2 WHERE {

  ?disease a :TreatableDisease;
           rdfs:label "Diabetes mellitus";
           :possibleDrug ?drug.

  ?drug rdfs:label ?label2;
        a :ApprovedDrug.

}
```

Fig 5. Query Example: Finding Possible Drugs

In this case the disease is identified through its label.

3.5.3 Listing diseases related by genes

We can look for disease that are affected by the same genes by querying the "relatedDisease" property. Figure 6 is a screencast of the query.

```
SELECT ?label1 ?label2 WHERE {

  ?disease1 :relatedDisease ?disease2;
            rdfs:label ?label1.
  ?disease2 rdfs:label ?label2.

}
```

Fig 6. Query Example: Listing Diseases Related by Genes

This can be used in a more advanced way by looking for specific genes.

4 USER DEFINITION

"Persona" is a novel method in user experience optimization, which means creating simulated models as representation of user segmentations. We applied Persona in this project to vividly explain functions of our application in perspective of real-life users.

4.1 User Segmentation

Both sides of a medical treatment are interested in the prescript drugs: healthcare providers and patients. Moreover, patients' family members and friends who care about them may also be involved. Figure 7 illustrates the estimated users segmentation of "InterDrugs"application.

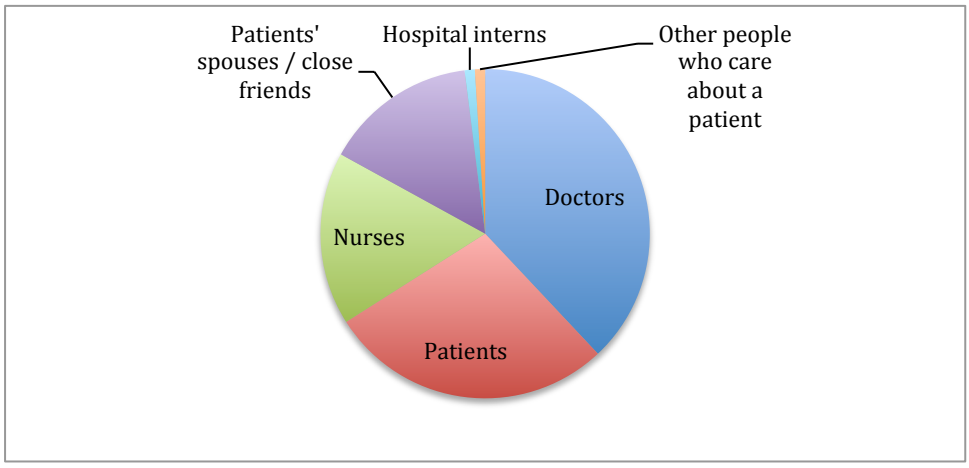

Fig 7. Users Segmentation (estimated[2]) for "InterDrugs" application

As no human being is capable to win love from everyone, it is impossible for any software to meet user needs of everybody. Considering this fact, we classified users into three levels and majorly focus on primary users. Table 2 shows our user priorities.

[2] In a standard process, the users segmentation should be results from real users study. Given limited time, we generated this result from literature investigation and brainstorming.

Tab 2. User Priorities for "InterDrugs" Application

Level	Identities	Priority in Conflicts
Primary Users	Doctors, Patients	First consideration in user needs conflicts.
Secondary Users	Nurses, Patients' Spouses/Close Friends	Fulfill their needs whenever possible.
Complementary Users	Hospital Interns, Other People Who Care About A Patient	Worth to consider in design.

4.2 Personas

We created two different personas for primary users: Doctor Nicole and Patient Edwin. Figure 8 depicts profiles of these two personas.

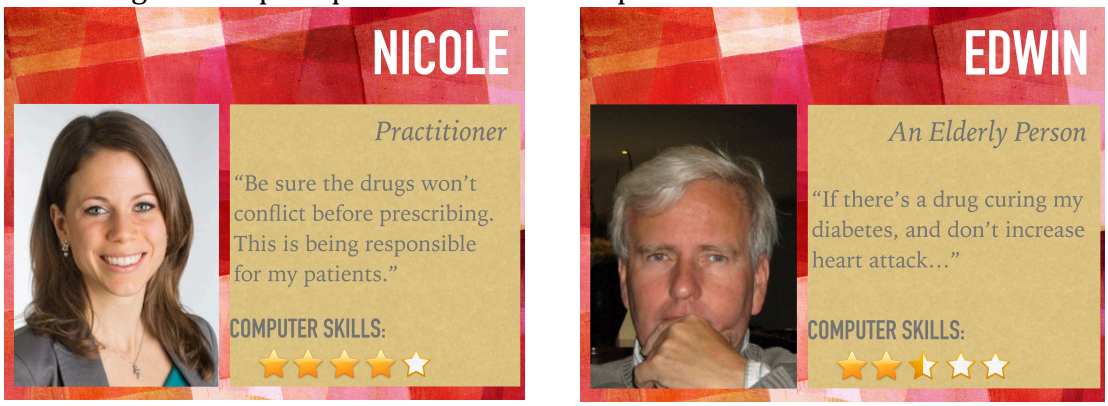

Fig 8. Personas of Primary Users for "InterDrugs" Application

4.2.1 Nicole (Doctor)

Nicole is a practitioner at an urban clinic. Diversity and mobility in this international city bring her challenges in healthcare job. She faces different people, various diseases. General knowledge on medicines equips her capability to bear responsibility for healthiness in the neighborhood, but she still needs some technical support to help her double-check the possible conflicts among drugs.

Nicole received high-quality education before working as a doctor; therefore she is quite familiar with Internet and computer operations. With the comparison columns of drugs and highlights of interacting ones, she is happy on the application's supporting when she decides prescriptions.

4.2.2 Edwin (Patient)

Edwin is retired and lives with his wife. Earlier career experiences nourished his strong ego, resulting in trying to get control of every aspect in his own life.

Suffering from diabetes for already 5 years, he has some basic understandings to the treatments. Recently he was detected a little bit heart valve disorders, and wants to learn some medical knowledge to guarantee his health, which is undoubtedly essential to life.

Edwin hates complicated computer settings. With the auto complete in typing disease names, he is satisfied of the application's usability.

5 INTERFACE DESIGN

5.1 Designed User Interface

Figure 9 shows our designed user interface. A screencast of the visual design prototype is at Vimeo (https://vimeo.com/142473832).

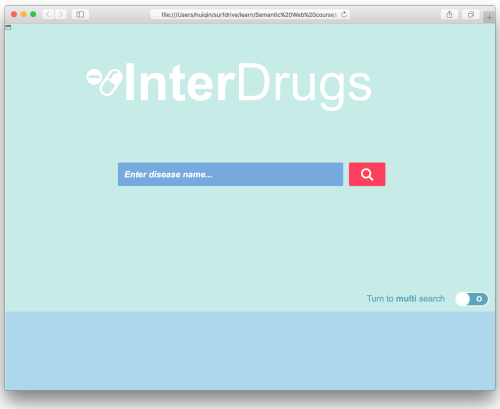

Fig 9. Designed User Interface

5.2 Realized User Interface

A screencast is at Vimeo (https://vimeo.com/143414010).