**Bachelor’s thesis**

TuberXpert

Not confidential

****

|  |  |
| --- | --- |
| **Student :** | **Melvyn Herzig** |
| **Work proposed by :** | Professor Yann Thoma  REDS  Route de Chesaux 1  1401 Yverdon-les-Bains |
| **Responsible teacher :** | **Professor Yann Thoma** |
| **Academic year :** | 2021-2022 |

Yverdon-les-Bains, the Sunday, February 24, 2022

Department TIC

Faculty Computer science

Specialization Software

Student Melvyn Herzig

Responsible teacher Prof. Yann Thoma

Bachelor’s thesis 2021-2022

Cliquez ici pour écrire votre titre de TB.

**Publishable summary**

Cliquez ou appuyez ici pour entrer du texte.

|  |  |  |
| --- | --- | --- |
| Student :  Herzig Melvyn | Date and time :  …………………………………… | Signature :  …………………………………… |
| Responsible teacher :  Thoma Yann | Date and time :  …………………………………… | Signature :  …………………………………… |

**Preamble**

This Bachelor's thesis (hereinafter referred to as "TB") is produced at the end of the course of study, with a view to obtaining the title of Bachelor of Science HES-SO in Engineering.

As an academic work, its content, without prejudging its value, does not engage the responsibility of the author, nor that of the jury of the Bachelor's thesis and of the school.

Any use, even partial, of this TB must be made in compliance with the copyright.

HEIG-VD

The head of the department

Yverdon-les-Bains, the Friday, March 4, 2022

**Authentication**

The undersigned, Melvyn Herzig, hereby certifies that he alone carried out this work and did not use any other source than those expressly mentioned.

Yverdon, the Friday, March 4, 2022

Melvyn Herzig

**Table of contents**

[1 Introduction 6](#_Toc98954809)

[1.1 What is therapeutic drug monitoring 6](#_Toc98954810)

[1.2 Actual situation in Tanzania 6](#_Toc98954811)

[1.3 Goal of this work 7](#_Toc98954812)

[2 Software needs 8](#_Toc98954813)

[2.1 Requirements 8](#_Toc98954814)

[2.1.1 Functional 8](#_Toc98954815)

[2.1.2 Nonfunctional 8](#_Toc98954816)

[2.2 Architecture 8](#_Toc98954817)

[3 Analysis 9](#_Toc98954818)

[3.1 Global application flow 9](#_Toc98954819)

[3.2 Technologies 9](#_Toc98954820)

[3.2.1 Development language 9](#_Toc98954821)

[3.2.2 Integrated Development Environment 9](#_Toc98954822)

[3.2.3 Compiler 10](#_Toc98954823)

[3.3 Input handling 10](#_Toc98954824)

[3.3.1 Global 10](#_Toc98954825)

[3.3.2 Covariates 11](#_Toc98954826)

[3.3.3 Drugs 11](#_Toc98954827)

[3.4 Drug file selection 12](#_Toc98954828)

[4 Implementation 12](#_Toc98954829)

[5 Tests 13](#_Toc98954830)

[6 Conclusion 14](#_Toc98954831)

# Introduction

## What is therapeutic drug monitoring

Nowadays, many drugs or antibiotics are used to treat diseases such as tuberculosis and HIV. Usually, the doctors prescribe generic doses that are suitable for the general population. Unfortunately, everyone’s metabolism reacts differently which makes generic dosages often ineffective.

Some people will have insufficient circulating drug exposure caused by an underdose. Thus, the treatment will be ineffective, and the patient may become drug resistant. Conversely, an overdose may result in intoxication. This would force an interruption of the treatment in order not to worsen the patient's health.

To avoid such situations, therapeutic drug monitoring (TDM) has been developed. TDM is a precision medicine that prescribes a personalized dosage to each patient based on the monitoring of the evolution of the drug concentration in the blood.

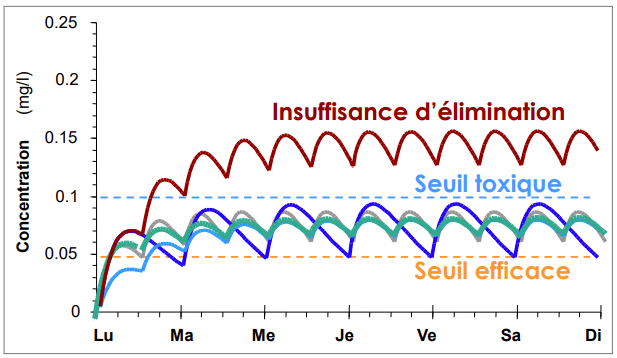


Figure 1 - Dosage Scheme (BUCLIN Thierry, 2022, Les Bases de la pharmacocinétique clinique [PDF document])

The purpose of TDM is to measure the concentration of the drug regularly and accurately in order to know its evolution. After examination, an expert in pharmacology can determine how to adjust the dosage to be above the threshold of inefficiency and below the threshold of toxicity.

## Actual situation in Tanzania

Tanzania has a high burden of Tuberculosis (TB). Over the last decade Tanzanian health authorities estimate an incidence of 120’000 – 150’000 patients per year for TB. The global community, through the END TB strategy, has declared its willingness to end TB by 2035, and a central component of the arsenal for this includes resorting to the correct use of anti-TB drugs, in particular the first-line agents: isoniazid, rifampicin, ethambutol, and pyrazinamide.

However, the problem is that only 60’000 – 75’000 patients are notified and receive a treatment. In addition, studies have reported that rifampicin dosages are insufficient most of the time. For example, investigations by Heysell et al. (2011) and Tostmann et al. (2013) in the Kilimanjaro region showed that one to two thirds of uncomplicated TB patients had maximum concentrations below the reference range of 8-24 mg/L, defined two hours after last dose intake.

Moreover, a significant proportion of individuals with TB are coinfected with HIV. It represents 25 to 40% of the monitored people. Administering antiretroviral drugs with first line antitubercular drugs lower furthermore the rifampicin concentration.

On top of that, TB patients have an increased risk to get affected by diabetes mellitus (DM). It represents 4-16% of the TB population. Unluckily DM may alter the pharmacokinetics (PK) of various drugs which include antitubercular. Mtabho et al. observed that DM predicted low levels of rifampicin in TB Tanzanian patients. Sadly, evidence have shown that individuals with TB and DM have a 5-fold risk of death compared to those without DM.

At this point, we easily understand that the risk of treatment failure or unfavorable outcome is real if the dosages are kept unsuitable.

## Goal of this work

The need to end tuberculosis is real and urgent in Tanzania. Unfortunately, TDM is a long and complicated process. In addition, the number of experienced pharmacologists is not sufficient to provide well established interpretation and recommendation everywhere their expertise is needed.

Currently, Professor Yann Thoma and his team have developed Tucuxi. It is a software intended for the practice of TDM. Already developed for several years, the software offers many features:

* Drug concentration predictions based on population and patient data (covariates) as well as on previous measurements.
* Suggestion of dosage adjustments to reach an optimal drug concentration state.
* Generation of printable reports.
* Integration with Electronics Health Record (EHR) systems.

Although it greatly simplifies the TDM process, Tucuxi is intended for experienced pharmacologists.

TuberXpert comes at the beginning of a large project between Switzerland and Tanzania led by Prof. Thoma Yann, Prof. Mpagama Stellah, and Prof. Mpagama Stellah. The aim is to develop a Clinical Decision Support System (CDSS) to fight tuberculosis. Thus, TuberXpert is a software layer that adds to the existing Tucuxi computing core. By receiving complete information from a patient, the system assesses the relevance of the data provided and then determines whether an adjustment of rifampicin dosage is necessary. All interpretations made by the program will then be provided to the user in the form of a simplified report compared to the original software. The main purpose of TuberXpert is to simplify the “Interpretation and recommendation” phase of TDM for non-experts

Figure 2 -TDM process (BUCLIN Thierry, 2022, Les Bases de la pharmacocinétique clinique [PDF document])

In other words, TuberXpert is a turnkey solution for rifampicin TDM. The software developed during this Bachelor's thesis is a first step. It will then be taken over by three PhD students in charge of the development and the concrete application of the project.

# Software needs

This chapter describes what TuberXpert needs to be and to do. All the information used to write this part are gathered from the SPIRIT SNSF project proposal document by Prof. THOMA Yann, prof. MPAGAMA Stellah, GUIDI Monia (see the first bibliographic reference).

## Requirements

### Functional

* Assess the relevance of query data.
* Chose a drug model to be used accordingly to the query.
* Assess the expectedness of a drug concentration result, considering the patient’s characteristics.
* Assess the target attainment of the current dosage.
* Propose a dosage adjustment if required.
* Present clear and meaningful messages within the report to help the clinician with the decision-making process.
* Generate alerts when data seems suspicious or erroneous.
* It can support various languages.

### Nonfunctional

* There is no need to develop a graphical user interface. A CLI is sufficient.
* The generated report must be user friendly and good looking.
  + The report to be generated shall not only display useful values and graphs but shall also offer readable sentences.
* The solution must be developed keeping in mind that it must be scalable. In other words, it is easy to be modified for other drugs compatible with Tucuxi or to change the behavior of the system.

## Architecture

In terms of software architecture, the clinical decision support system (CDSS) may be separated from the report generation. The CDSS will offer a standardized output (most probably in XML format). This output will be used by the report generator to fill some fields of a report template.

# Analysis

## Global application flow

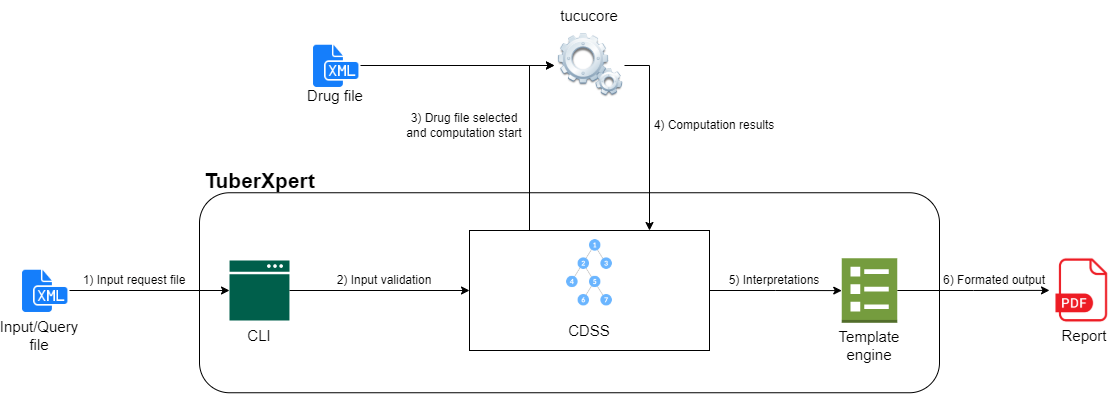


Figure 3 Global application flow and components

1. The program receives as input a query in XML format. For simplification purpose, the XML format will be the same used by tucucore.
2. The program loads the data from the query file and transmit them to the CDSS that will determine if the data are consistent.
3. The CDSS will determine the best drug model to be used and starts the computation.
4. The computations results are transmitted to the CDSS which analyses them.
5. The decisions are passed to a template engine that format them into a readable format.
6. The report is saved in a PDF file.

## Technologies

### Development language

Initially, Tucuxi was developed in C++.This language was chosen for its good performance since the software requires a lot of computing performance. Then, other projects were added. Most of them were also developed in C++ for the same reasons.

Thus, to preserve the homogeneity of the projects, TuberXpert will also be developed in C++. This language is once again very advantageous. As the software will probably be used on low-powered computers in Tanzania, we will optimize the performance by using a low-level language.

Version: C++ 20

### Integrated Development Environment

Once again, we will walk in the steps of Tucuxi. We will use the same development environment . QT is a very advantageous and rich IDE. It easily allows cross-platform development, test management, language management and many other frameworks. QT is very versatile. It also makes it easier to work with existing projects at the same time. De facto it is the IDE that will be used to develop this project.

Version: QT 6.0.2 community

### Compiler

To simplify the installation of the development environment, this project is compiled with the compiler integrated with QT.

Version (Windows): MinGW-W64-builds-5.0.0

## Input handling

The input will be an XML file with the same structure as tucucore. For a complete input specification, see the file “Tucuxi CLI Software Usability Specification”. Consequently, this chapter will not explain the expected structure but the checks that will be made by the system.

All abnormal data will produce a warning in the final report.

**The program should be easily scalable to add rules of check.**

### Global

**Example of the global structure of the input**:

<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<query version="1.0"

    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

    xsi:noNamespaceSchemaLocation="computing\_query.xsd">

    <queryId>imatinib\_2</queryId>

    <clientId>124568</clientId>

    <date>2018-07-11T13:45:30</date>

    <language>en</language>

    <drugTreatment>

        <patient>

            <covariates></covariates>

        </patient>

        <drugs></drugs>

    </drugTreatment>

    <requests></requests>

</query>

**Checks :**

None.

### Covariates

The covariates element contains zero or many covariate elements .

**Example of covariate:**

<covariate>

    <covariateId>bodyweight</covariateId>

    <date>2018-07-11T10:45:30</date>

    <value>40</value>

    <unit>kg</unit>

    <dataType>double</dataType>

    <nature>discrete</nature>

</covariate>

**Checks:**

There is no exhaustive list of covariates. Therefore, in the context of this work, it is difficult to produce a system that handles all of them.

We can imagine two types of checks:

* Global: Made on every covariate.
* Single: Made on a specific covariate determined by covariateId.

The project will at least check the following points:

* Global: Check the date. If the date is older than one year, display a warning.
* Single: birthdate, between the current date and current date minus 110 years.
* Single: bodyweight, between 0 and 100.
* Single: Fat free mass, between 0 and 100.
* Single: sex, 0 or 1.

Faire confiance à la fiche, laisser tomber ces verification

### Drugs

The drugs element contains some drug elements about treatments the patient could be in.

**Example of drug:**

<drug>

    <drugId>imatinib</drugId>

    <activePrinciple>something</activePrinciple>

    <brandName>somebrand</brandName>

    <atc>something</atc>

    <treatment>

    </treatment>

    <samples></samples>

    <targets></targets>

</drug>

**Checks:**

Check if the drug id corresponds to at least one drug file.

#### Treatment

It contains the dosage history of the patient for a specific drug.

<treatment>

    <dosageHistory>

        <dosageTimeRange>

            <start>2018-07-06T08:00:00</start>

            <end>2018-07-08T08:00:00</end>

            <dosage></dosage>

        </dosageTimeRange>

    </dosageHistory>

</treatment>

**Checks:**

When there are multiple *dosageTimeRange*, check if they overlap.

The core already checks if the date start is smaller than date end. In that case, the computation fails.

#### Dosage

The dosage element is complex. For the sake if simplicity, only a few rifampin-based controls will be performed.

Check if the value is not extravagant: Unit in mg and value not exceeding 2000 (voir fiche medicament)

## Drug file selection

One task of the system is to choose a drug file to use. To do this, a simple heuristic will be implemented. Once again, the program should be easily modified to implement a more elaborate method.

**Heuristic behavior**

For each available model of a given drug, we calculate a score S based on the covariates of the model.

*S = +*

The model with the lowest score is chosen. In case of a tied, the model with the most respected covariates is chosen.

**Comments**

Again, this method is far from optimal, but it is a good starting point. What happens if a covariate is mandatory or if two models tied perfectly? This type of question is not considered by this algorithm. In the future, a close collaboration with doctors will be necessary in order to determine for each drug “how to choose the drug file that fit the most to the patient”.

# Implementation

# Tests

# Conclusion

**Bibliography**

* Prof. THOMA Yann, prof. MPAGAMA Stellah, GUIDI Monia, 2022, TuberXpert: Clinical Decision Support System for antituberculosis medical drugs [PDF document]
* BENALLAL Nadir, 2018, Expert System for drug dosage adaptation [PDF document], Yverdon-les-Bains : Haute école d’ingénierie et de gestion du canton de Vaud. Bachelor’s thesis
* CADUFF Max, 2020, Expert System for drug dosage adjustments [PDF document], Yverdon-les-Bains : Haute école d’ingénierie et de gestion du canton de Vaud. Bachelor’s thesis
* BUCLIN Thierry, 2022, Les Bases de la pharmacocinétique clinique [PDF document]
* TUCUXI dev team, 2022, Tucuxi CLI Software Usability Specification [PDF document]
* TUCUXI dev team, 2022, Tucuxi Drug Model File Specification [PDF document]
* DATAPHARM, 2022, EMC, 21 Feb 2022 [24 Mar 2022], https://www.medicines.org.uk/emc/product/8788/smpc