Architecture Design

TI2806 Contextproject
Health Informatics

Group A

Technische Universiteit Delft

TU Delft Delft University of Technology

Challenge the future

CONTENTS

1

INTRODUCTION

This document describes the architecture of the application. Primarily it will provide a high level description of the structure of the software.

1.1. DESIGN GOALS

A number of design goals have been establish for the application.

- **Maintainability**
  No matter how robust an application is, it *will* have to be maintained after its initial deployment for a variety of reasons. For example, defects that did not emerge during testing, changes in the client's environment or even changes in the client's requirements. It is therefor important that the system is designed in a manner such that it can be easily maintained.

- **Testability**
  Proper testing allows defects to be found and fixed with relative ease. Therefor the system should be designed in such a way that it allows for effective testing. This can be realized through modular design. The system should not only be modular on a high level, but also on unit level. For example, through the use of interfaces to decrease coupling between units.

2

SOFTWARE ARCHITECTURE VIEWS

2.1. SUBSYSTEM DECOMPOSITION

The system is divided into four main packages, each with a specific and well defined responsibility. This section will provide an overview of the different subsystems.

- **Control Module**
  The control module is effectively the driving package behind the application. It parses and interprets a script provided by the user to determine which actions should be performed. It then asks the input module to parse the files specified by the user. Then it will ask the analysis module to perform all analysis methods specified in the script on the parsed data. Finally it tells the output module to generate visualizations and formatted files as specified in the script.
- **Input Module**
  The input module will parse input data provided by the user into a software representation that all modules can work with. Input files will require an XML file that describes the layout of the input file, so that if the user would like to use other data, only the XML description file has to be modified.
- **Analysis Module**
  The analysis module provides important transformations that are commonly needed in sequential data analysis.
- **Output Module**
  The output module can generate certain visualizations based on the analyzed data and is able to output the data to a file in a format as specified by the user.

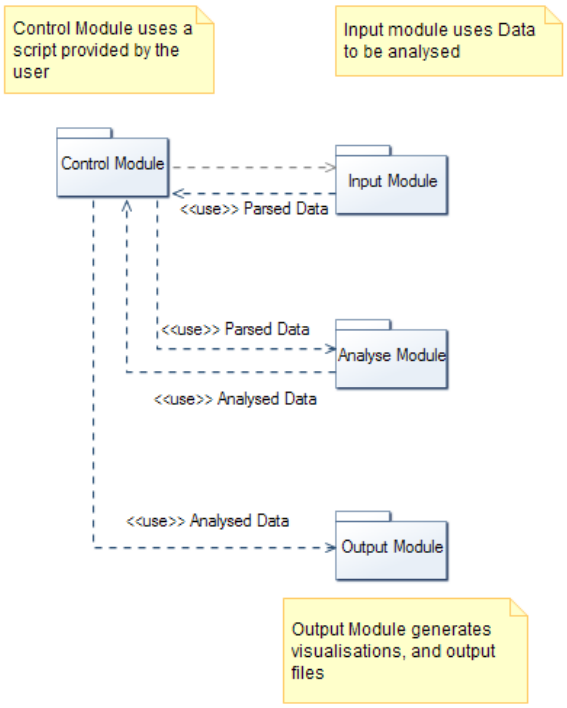

Figure 2.1: The subsystems and the data flow between them

2.2. HARDWARE/SOFTWARE MAPPING

<todo>

2.3. PERSISTENT DATA MANAGEMENT

<todo>

2.4. CONCURRENCY

<todo>

3

GLOSSARY

<todo>