

# COSBAS Architectural Requirements Documentation

Git: <https://github.com/undecidables/Requirements-Documentation>

**<undecidables>**

Elzahn Botha *u13033922*

Jason Richard Evans *u13032608*

Renette Ros

Szymon Ziolkowski

Tienie Pritchard

Vivian Venter

**March 2015**

# Contents

<b>1</b>	<b>Architectural Requirements</b>	<b>2</b>
1.1	Introduction . . . . .	2
1.2	Architectural Scope . . . . .	2
1.3	Quality Requirements . . . . .	2
1.4	Integration and Access Channel Requirements . . . . .	2
1.4.1	Human access channels . . . . .	2
1.4.2	System access channels . . . . .	2
1.4.3	Integration Channels . . . . .	2
1.5	Architecture Constraints . . . . .	3
<b>2</b>	<b>Architectural Patterns or Styles</b>	<b>3</b>
2.1	MVC Architectural Pattern . . . . .	3
2.1.1	Description . . . . .	3
2.1.2	Reason for use . . . . .	3
2.2	Adapter Design Pattern . . . . .	3
2.2.1	Description . . . . .	3
2.2.2	Reason for use . . . . .	3
<b>3</b>	<b>Architectural Tactics or Strategies</b>	<b>3</b>
<b>4</b>	<b>Use of Reference Architectures and Frameworks</b>	<b>3</b>
<b>5</b>	<b>Access and Integration Channels</b>	<b>3</b>
<b>6</b>	<b>Technologies</b>	<b>3</b>
6.1	Hardware . . . . .	3

# **1 Architectural Requirements**

## **1.1 Introduction**

The software architecture requirements for the COSBAS system.

## **1.2 Architectural Scope**

## **1.3 Quality Requirements**

## **1.4 Integration and Access Channel Requirements**

### **1.4.1 Human access channels**

This system will be accessible to humans in the followings ways:

- From a thin client(can be computer with the client program but in this case it will be a Raspberry Pi) which will be installed at each entrance/exit of the building through non-intrusive bio-metrics or keypad.

### **1.4.2 System access channels**

The client(can be computer with the client program but in this case it will be a Raspberry Pi) should be able to access the services provided by the system to authenticate a user who would like to enter or exit the building. This will be done through SOAP based web services.

### **1.4.3 Integration Channels**

The following integration channels will be made use of for the creation of the COSBAS system.

- The CS LDAP server in order to retrieve login details of the lecturers.
- The postgrad meeting system in order to help with making appointments.
- Any online calendars used, such as google calendar or outlook, in order to gain access to the lecturers' calendars.
- The COSBAS-server to process the data and grant or deny access.
- The spring MVC framework, to help with dependency injection and connecting all the componenets together.

## **1.5 Architecture Constraints**

# **2 Architectural Patterns or Styles**

## **2.1 MVC Architectural Pattern**

### **2.1.1 Description**

MVC (Model-view-controller) is a software architectural pattern which divides the software application into three interconnected parts, so as to separate the internal representation from the way the information is represented to the user.

### **2.1.2 Reason for use**

- Client-Server communication
- Reduced code complexity
- Efficient code-reuse
- Decoupled code

## **2.2 Adapter Design Pattern**

### **2.2.1 Description**

The adapter design pattern changes or converts the interface of a class into another interface the client expects. The design pattern makes classes that would normally not be able to work together, interact seamlessly.

### **2.2.2 Reason for use**

- Increased pluggability of the system - Because many different biometric access points as well as non-biometric access points will have to interact with the system. This makes it easy for a new type of access point to be added to the system.

# **3 Architectural Tactics or Strategies**

# **4 Use of Reference Architectures and Frameworks**

# **5 Access and Integration Channels**

# **6 Technologies**

## **6.1 Hardware**