

# Contents

<b>1</b>	<b>Dojo use case</b>	<b>3</b>
1.1	Unidentified user . . . . .	3
1.1.1	Overview . . . . .	3
1.1.2	Get informed . . . . .	3
1.1.3	Register . . . . .	3
1.1.4	Authenticate . . . . .	3
1.2	Student . . . . .	4
1.2.1	Overview . . . . .	4
1.2.2	Practice . . . . .	4
1.2.3	Meditate . . . . .	5
1.3	Teacher . . . . .	6
1.3.1	Overview . . . . .	6
1.3.2	Supervise students . . . . .	6
1.3.3	Observe . . . . .	8
1.4	Administrator . . . . .	9
1.4.1	Overview . . . . .	9
1.4.2	Supervise the platform . . . . .	9
1.5	Common registered user . . . . .	10
1.5.1	Communicate . . . . .	10
<b>2</b>	<b>Dojo diagram package</b>	<b>11</b>
2.1	Overview . . . . .	11
2.2	User Interface . . . . .	12
2.2.1	Web . . . . .	12
2.2.2	Unix . . . . .	12
2.3	HTTP . . . . .	12
2.4	Core . . . . .	12
2.4.1	Exercise . . . . .	12
2.4.2	Machinist . . . . .	12
2.4.3	Reactive . . . . .	12
2.4.4	User . . . . .	12
2.4.5	Client . . . . .	12
2.4.6	Errors . . . . .	12
2.5	Common . . . . .	12

<b>3</b>	<b>Diagram class</b>	<b>13</b>
3.1	User Interface . . . . .	13
3.1.1	Web . . . . .	13
3.2	HTTP . . . . .	13
3.3	Core . . . . .	14
3.3.1	Exercise . . . . .	14
3.3.2	Exercise : Description . . . . .	15
3.3.3	Machinist . . . . .	16
3.3.4	Reactive . . . . .	16
3.3.5	Reactive : Identifier . . . . .	17
3.3.6	Reactive : Client . . . . .	18
3.3.7	User . . . . .	18
3.3.8	Config . . . . .	19
3.3.9	VFS (Versioned File System) . . . . .	19
3.3.10	Common . . . . .	20
3.3.11	Common : Errors . . . . .	20
3.4	Common . . . . .	21

# Chapter 1

## Dojo use case

### 1.1 Unidentified user

#### 1.1.1 Overview

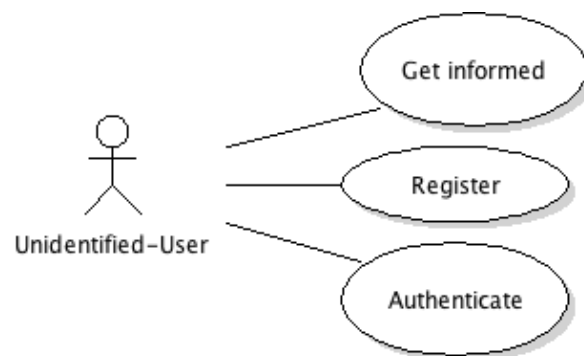


Figure 1.1: Unidentified user Use Case : Overview

#### 1.1.2 Get informed

An unidentified user gets inform about the platform.

#### 1.1.3 Register

An unidentified user who wants to access to the features of the platform has to register first.

#### 1.1.4 Authenticate

An unidentified user authenticates to have access to the features is already registered.

## 1.2 Student

### 1.2.1 Overview

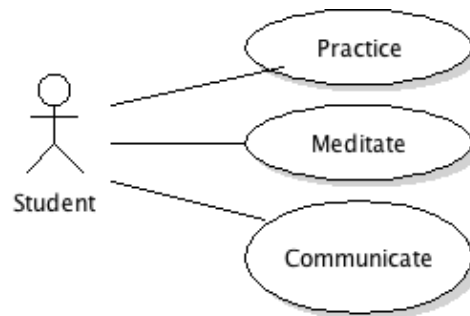


Figure 1.2: Student Use Case : Overview

### 1.2.2 Practice

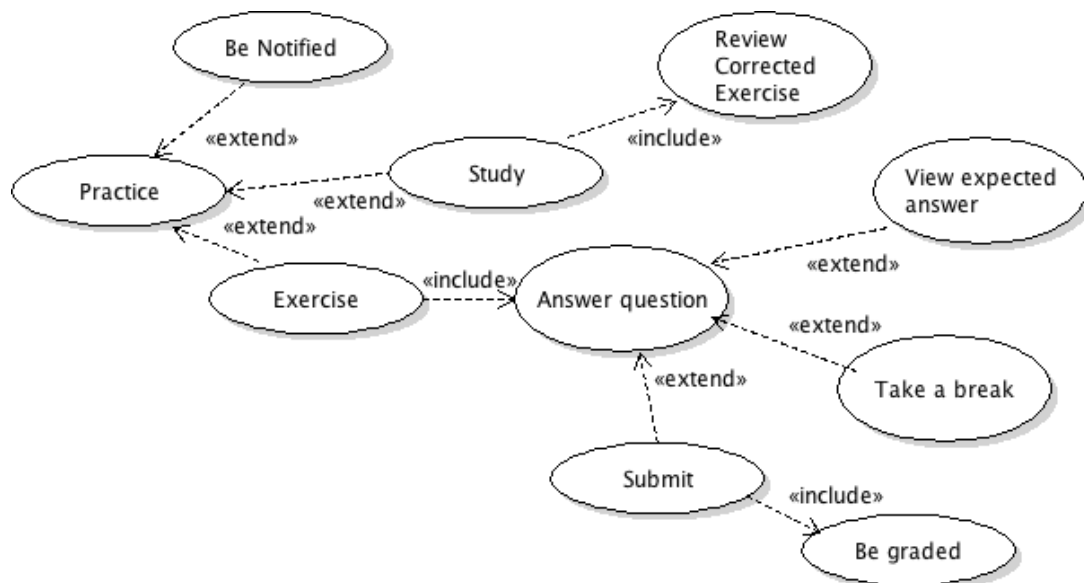


Figure 1.3: Student Use Case : Practice

#### Be notified

The student is notified for new events.

#### Study

The student studies by reviewing corrected exercises.

### **Exercise**

The student exercises by answering questions.

### **Answer question**

The student answers a set of question.

### **View expected answer**

The student has optionally access to the corrected version.

### **Take a break**

The student takes a break and resume the exercise later.

### **Submit**

The student can submit his answer form.

### **Be graded**

The student is also graded by the platform AFTER he submits his answer form.

## **1.2.3 Meditate**

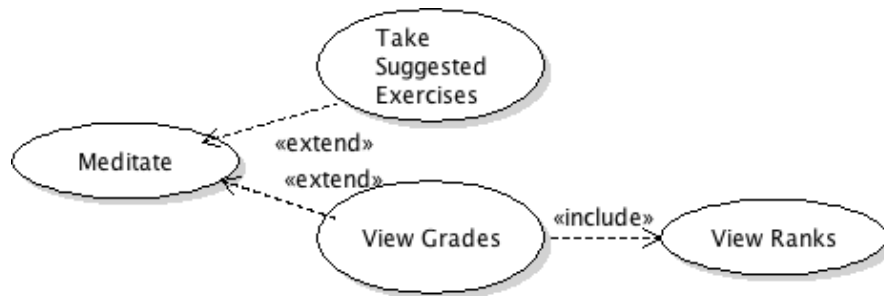


Figure 1.4: Student Use Case : Meditate

### **Take suggested exercises**

The student access and starts a set of exercises according to their grade.

### **View Grades**

The student can see their grade for each exercises.

### **View Ranks**

The student can compare himself with other students.

## 1.3 Teacher

### 1.3.1 Overview

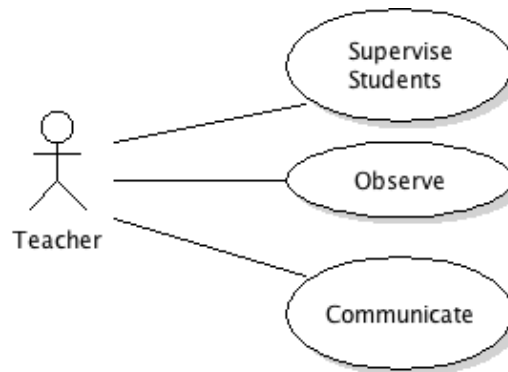


Figure 1.5: Teacher Use Case : Overview

### 1.3.2 Supervise students

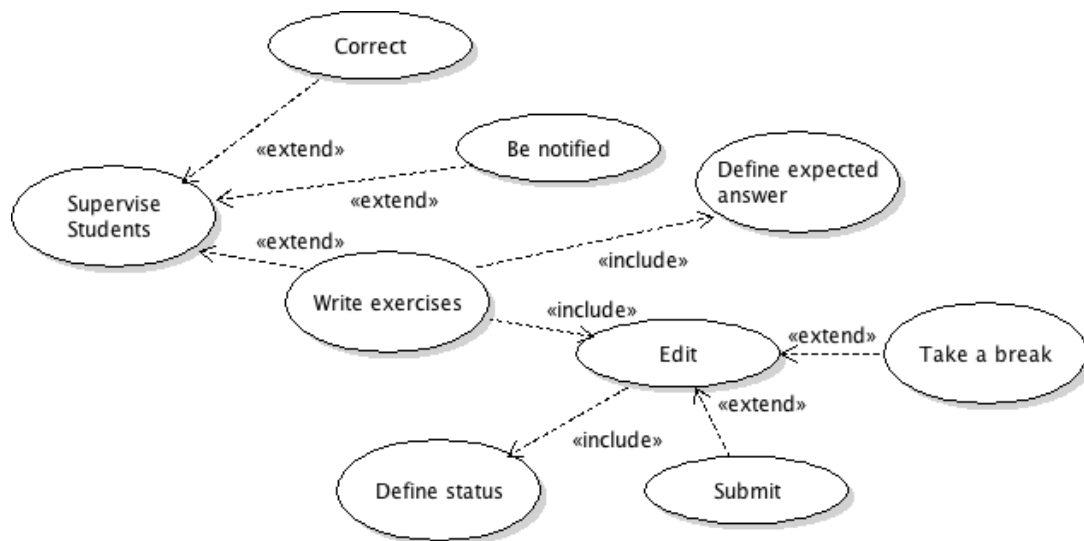


Figure 1.6: Teacher Use Case : Supervise students

#### Correct

The teacher has to grade students answers for questions which have no expected answers.

**Be Notified**

The teacher will be notified of events. For instance the teacher will be notified when one of his exercises is online.

**Write exercises**

The teacher supervises his students by writing exercises.

**Define expected answer**

When the teacher writes exercises, he can optionally define an expected answer.

**Edit**

The teacher can edit any exercise at anytime.

**Define Status**

The teacher attaches a status to an exercise.

**Take a break**

The teacher can resume his exercise later.

**Submit**

The teacher submits his exercise which can be viewed by student according to the exercise status defined by the teacher.

### 1.3.3 Observe

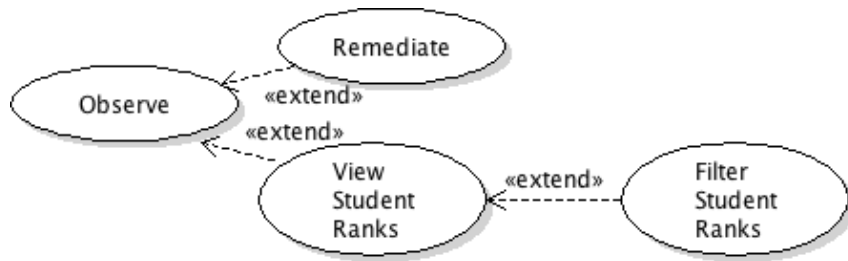


Figure 1.7: Teacher Use Case : Observe

#### **Rank**

The teacher sees all the grades of students who took his exercises.

#### **Filter**

The teacher can filter the grades according to some parameters. The parameters can be the identifier of a group, the year for instance.

#### **Remediate**

The teacher can provide exercises for a group of students in trouble with some specific subject of the course according to their grades.



## 1.4 Administrator

### 1.4.1 Overview

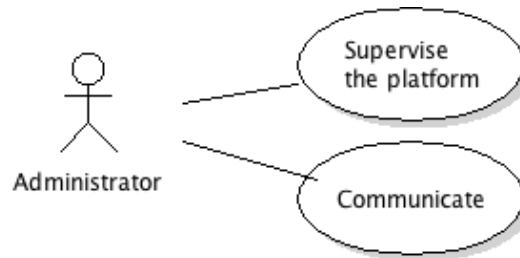


Figure 1.8: Administrator Use Case : Overview

### 1.4.2 Supervise the platform

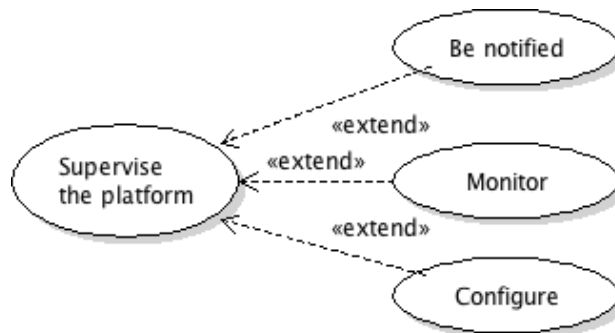


Figure 1.9: Administrator Use Case : Supervise the platform

#### **Be notified**

The administrator will be notified by events such as critical issue.

#### **Monitor**

The administrator inspects the log.

#### **Configure**

The administrator configures the platform.

## 1.5 Common registered user

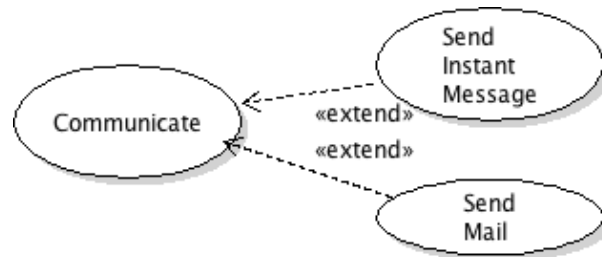


Figure 1.10: Registered user Use Case : Communicate

### 1.5.1 Communicate

#### Send Mail

Identified user communicate by mail exchange.

#### Send Instant Message

Identified user communicate by instant message exchange.

## Chapter 2

# Dojo diagram package

### 2.1 Overview

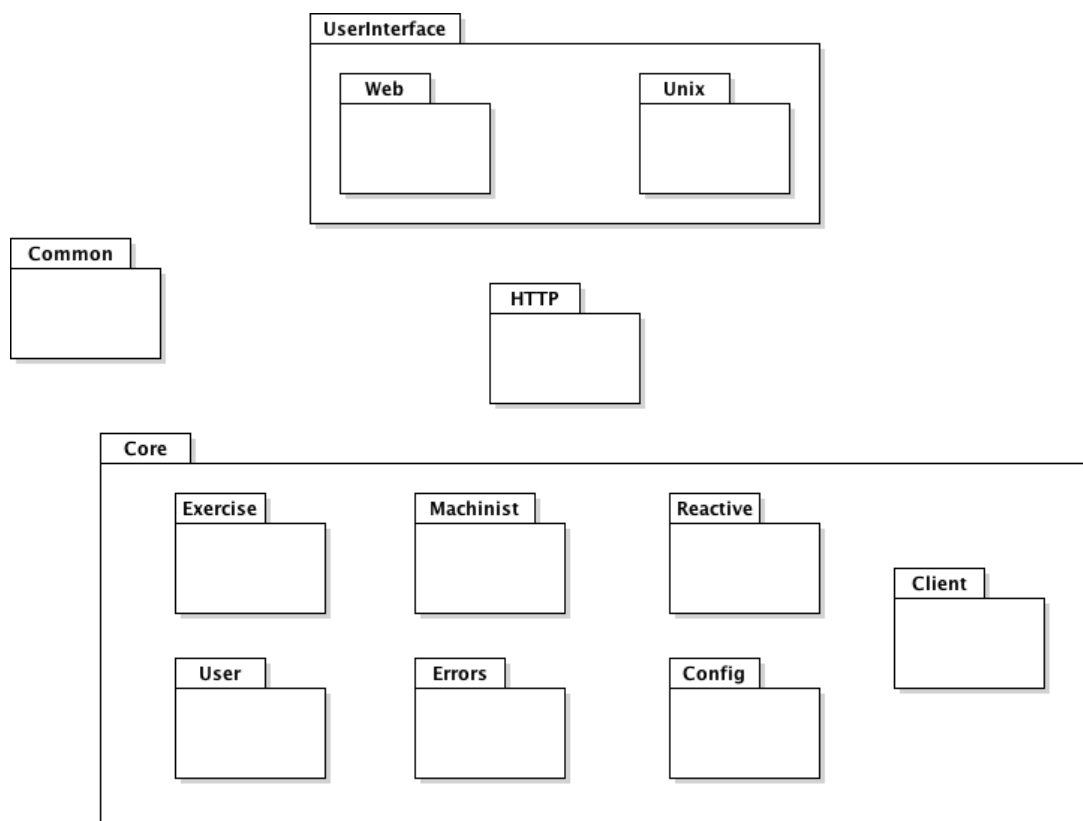


Figure 2.1: Diagram package : Overview

## **2.2 User Interface**

This package holds every interface for the user.

### **2.2.1 Web**

This package holds units which define the Web Interface

### **2.2.2 Unix**

This package holds units which define a Shell Interface for Unix user.

## **2.3 HTTP**

This package holds the web services which allow the user interface to communicate with the core.

## **2.4 Core**

This package holds the logical platform units.

### **2.4.1 Exercise**

This package holds units which define exercise entities.

### **2.4.2 Machinist**

This package holds units which define machinist entities.

### **2.4.3 Reactive**

This package holds units which define reactive entities.

### **2.4.4 User**

This package holds units which define user entities.

### **2.4.5 Client**

This package holds units which define client.

### **2.4.6 Errors**

This package holds units which define errors entities.

## **2.5 Common**

This package holds every units which are used in each package.

## Chapter 3

# Diagram class

### 3.1 User Interface

#### 3.1.1 Web

### 3.2 HTTP

## 3.3 Core

### 3.3.1 Exercise

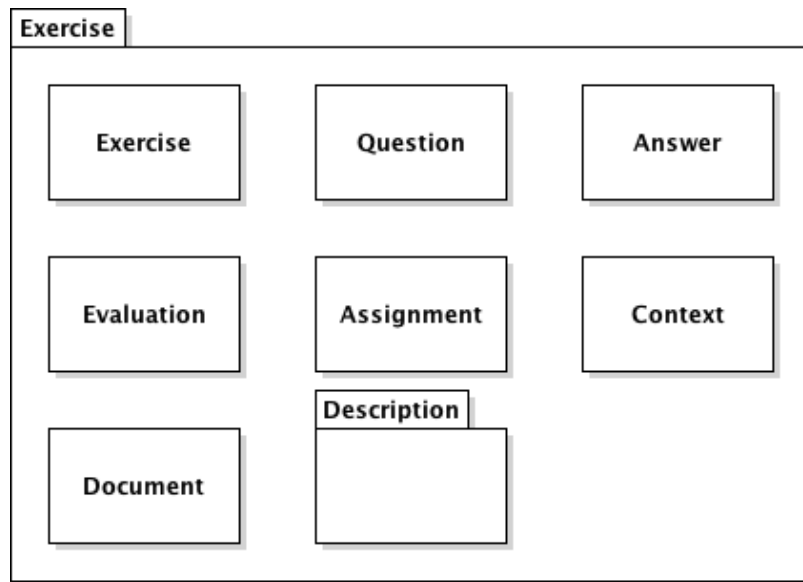


Figure 3.1: Core diagram class : Exercise

#### **Exercise**

This module implements an exercise entity.

#### **Question**

???

#### **Answer**

This module implements an answer entity part of a question

#### **Evaluation**

This module implements an evaluation entity on exercises.

#### **Assignment**

This module implements a priority entity on exercise for students.

#### **Context**

This module implements a context for an evaluation.

## Document

???

### 3.3.2 Exercise : Description

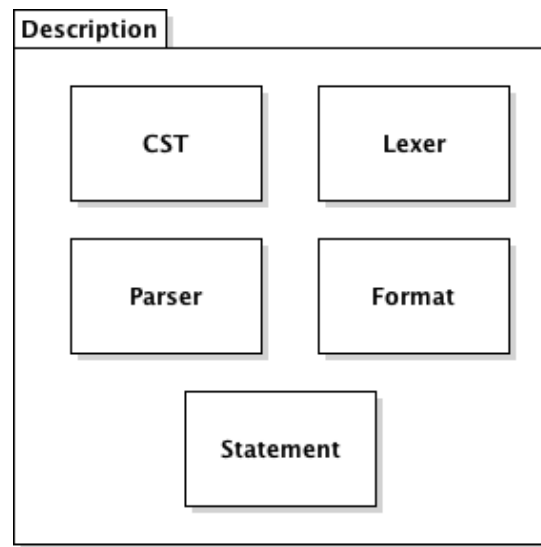


Figure 3.2: Core diagram class : Exercise - Description

#### CST

This module implements the syntax rules for a statement.

#### Lexer

This module implements rules for the lexer.

#### Parser

This module implements rules for the parser.

#### Format

This module implements a parser for a description.

#### Statement

???

### 3.3.3 Machinist

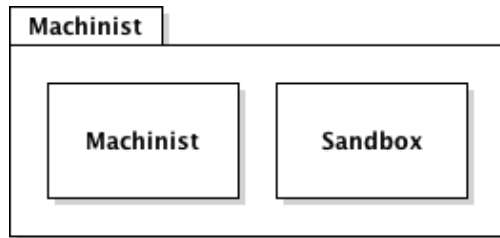


Figure 3.3: Core diagram class : Machinist

#### Machinist

This module implements a sandbox provider.

#### Sandbox

This module implements an abstraction on environment used for the sandbox.

### 3.3.4 Reactive

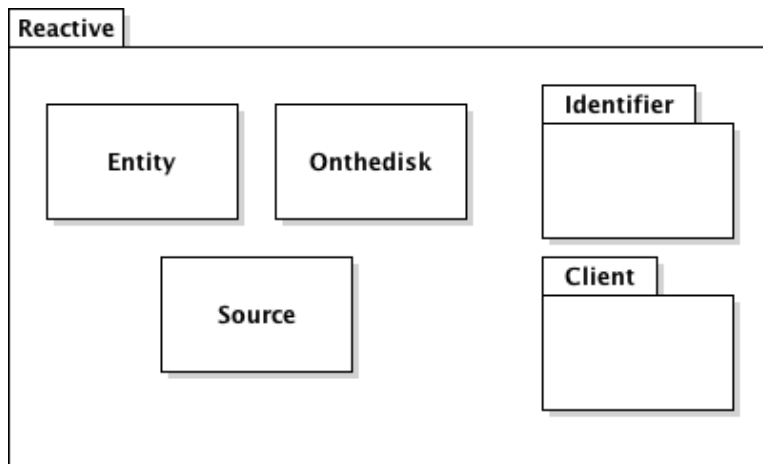


Figure 3.4: Core diagram class : Reactive



### **Entity**

This module implements a reactive entity.

### **Onthedisk**

This module implements an entity state to be serialized.

### **Source**

This module implements a system which associates a filename with a content.

## **3.3.5 Reactive : Identifier**

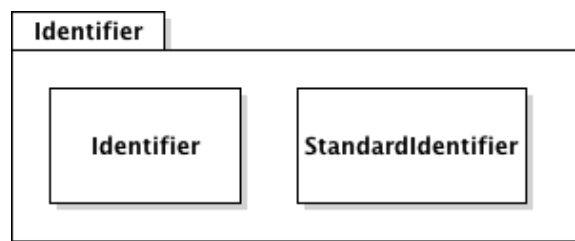


Figure 3.5: Core diagram class : Reactive - Identifier

### **Identifier**

This module implements an identifier for a reactive entity.

### **StandardIdentifier**

This module implements a standard identifier for reactive entities.

### 3.3.6 Reactive : Client

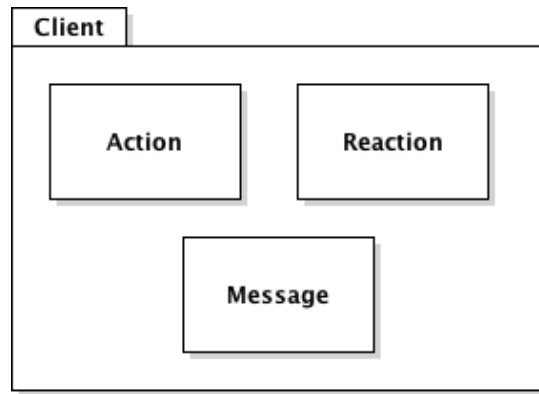


Figure 3.6: Core diagram class : Reactive - Client

#### Action

This module implements an association with a process and a client.

#### Reaction

This module implements a communication channel between the server and the client.

### 3.3.7 User

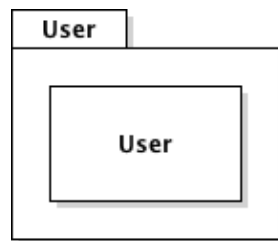


Figure 3.7: Core diagram class : User

#### User

This module implements an user entity.

### 3.3.8 Config

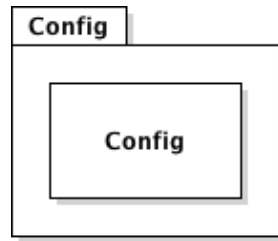


Figure 3.8: Core diagram class : Config

#### Config

This unit implements a set of general rules affecting the server behavior.

### 3.3.9 VFS (Versioned File System)

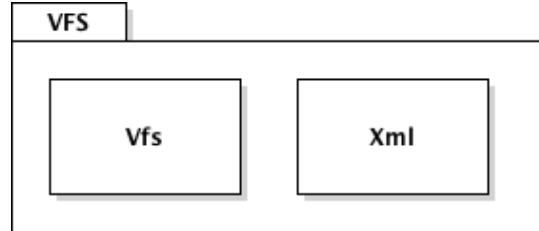


Figure 3.9: Core diagram class : VFS

#### VFS

This module implements a versioned hierarchical file system.

#### XML

This unit implements a XML parser.

### 3.3.10 Common

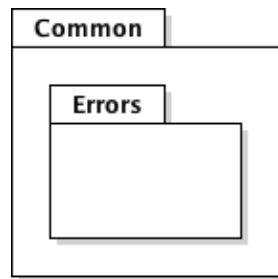


Figure 3.10: Core diagram class : Common

### 3.3.11 Common : Errors

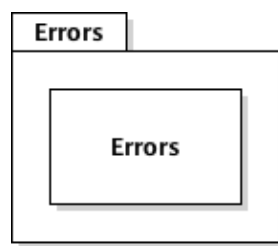


Figure 3.11: Core diagram class : Common - Errors

### Errors

This module implements type of exception.

### 3.4 Common

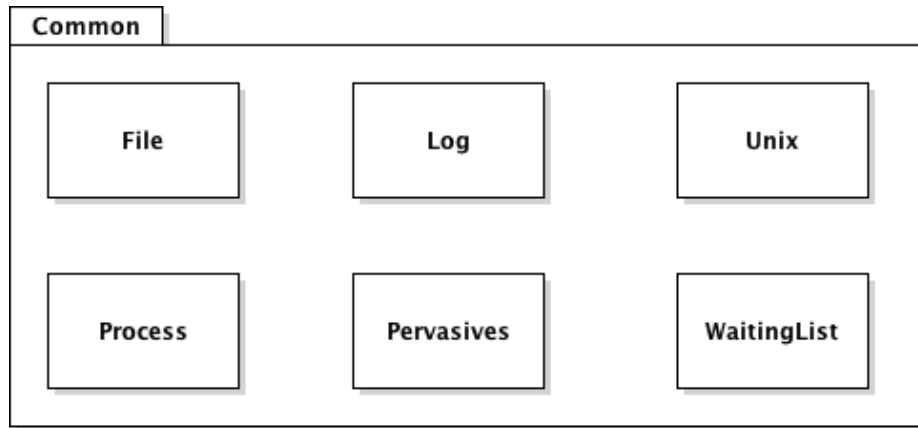


Figure 3.12: Common diagram class : Overview

# List of Figures

1.1	Unidentified user Use Case : Overview . . . . .	3
1.2	Student Use Case : Overview . . . . .	4
1.3	Student Use Case : Practice . . . . .	4
1.4	Student Use Case : Meditate . . . . .	5
1.5	Teacher Use Case : Overview . . . . .	6
1.6	Teacher Use Case : Supervise students . . . . .	6
1.7	Teacher Use Case : Observe . . . . .	8
1.8	Administrator Use Case : Overview . . . . .	9
1.9	Administrator Use Case : Supervise the platform . . . . .	9
1.10	Registered user Use Case : Communicate . . . . .	10
2.1	Diagram package : Overview . . . . .	11
3.1	Core diagram class : Exericse . . . . .	14
3.2	Core diagram class : Exercise - Description . . . . .	15
3.3	Core diagram class : Machinist . . . . .	16
3.4	Core diagram class : Reactive . . . . .	16
3.5	Core diagram class : Reactive - Identifier . . . . .	17
3.6	Core diagram class : Reactive - Client . . . . .	18
3.7	Core diagram class : User . . . . .	18
3.8	Core diagram class : Config . . . . .	19
3.9	Core diagram class : VFS . . . . .	19
3.10	Core diagram class : Common . . . . .	20
3.11	Core diagram class : Common - Errors . . . . .	20
3.12	Common diagram class : Overview . . . . .	21