

---

# Platform for Control and Delivery of services in Next Generation Networks

***DELIVERABLE 3.2***

**Prototype**



---

PROPRIETARY INFORMATION

©CEFRIEL 2008 All Rights Reserved

©PoliTO 2008 All Rights Reserved

This document and the data included in this document is proprietary to CEFRIEL and Torino's polytechnic, and is not to be reproduced, used, or disclosed in whole or in part to anyone without the express written permission of the parts above. The content of this document is provided for informational use only and is subject to change without notice.

Questions about this document or the features it describes should be directed to:

CEFRIEL

---

---

**RECORD OF CHANGES**

<b>Version</b>	<b>Date</b>	<b>Description</b>
0.1	30/10/2009	First draft version
0.2	09/11/2010	Second draft revision
1.0	31/01/2011	Final version

---

---

# Index

---

<b>Index .....</b>	<b>4</b>
<b>1 Introduction.....</b>	<b>6</b>
<b>2 Persistence .....</b>	<b>7</b>
2.1 Persistent data and Dynamic data .....	7
<b>3 Prototype Design .....</b>	<b>9</b>
3.1 PICO Server (PSCDS) .....	9
3.1.1 Technologies .....	9
3.1.2 Subsystems.....	11
3.1.3 Source code structure and packages.....	12
3.1.4 General Class Diagram .....	13
3.2 PICO Client (PSCD) .....	14
3.2.1 Technologies .....	14
3.2.2 Subsystems.....	15
3.2.3 Source code structure and packages.....	16
<b>4 Prototype Scenario.....</b>	<b>17</b>
4.1 Storyboard .....	17
<b>5 User Interface .....</b>	<b>19</b>
5.1 Introduction .....	19
5.2 User interface and visual design .....	19
<b>Table of Acronyms.....</b>	<b>Errore. Il segnalibro non è definito.</b>
<b>Index of Figures.....</b>	<b>21</b>

---

---

<b>6</b>	<b>Bibliography</b> .....	Errore. Il segnalibro non è definito.
----------	---------------------------	---------------------------------------

---

# 1 Introduction

---

This document shows in detail the architecture and the implementation of the prototype. After a brief description of persistence data, the document describes two different sections: the server and client side. Afterwards PICO Client will be analyzed in terms of user interface.

---

## 2 Persistence

---

### 2.1 Persistent data and Dynamic data

---

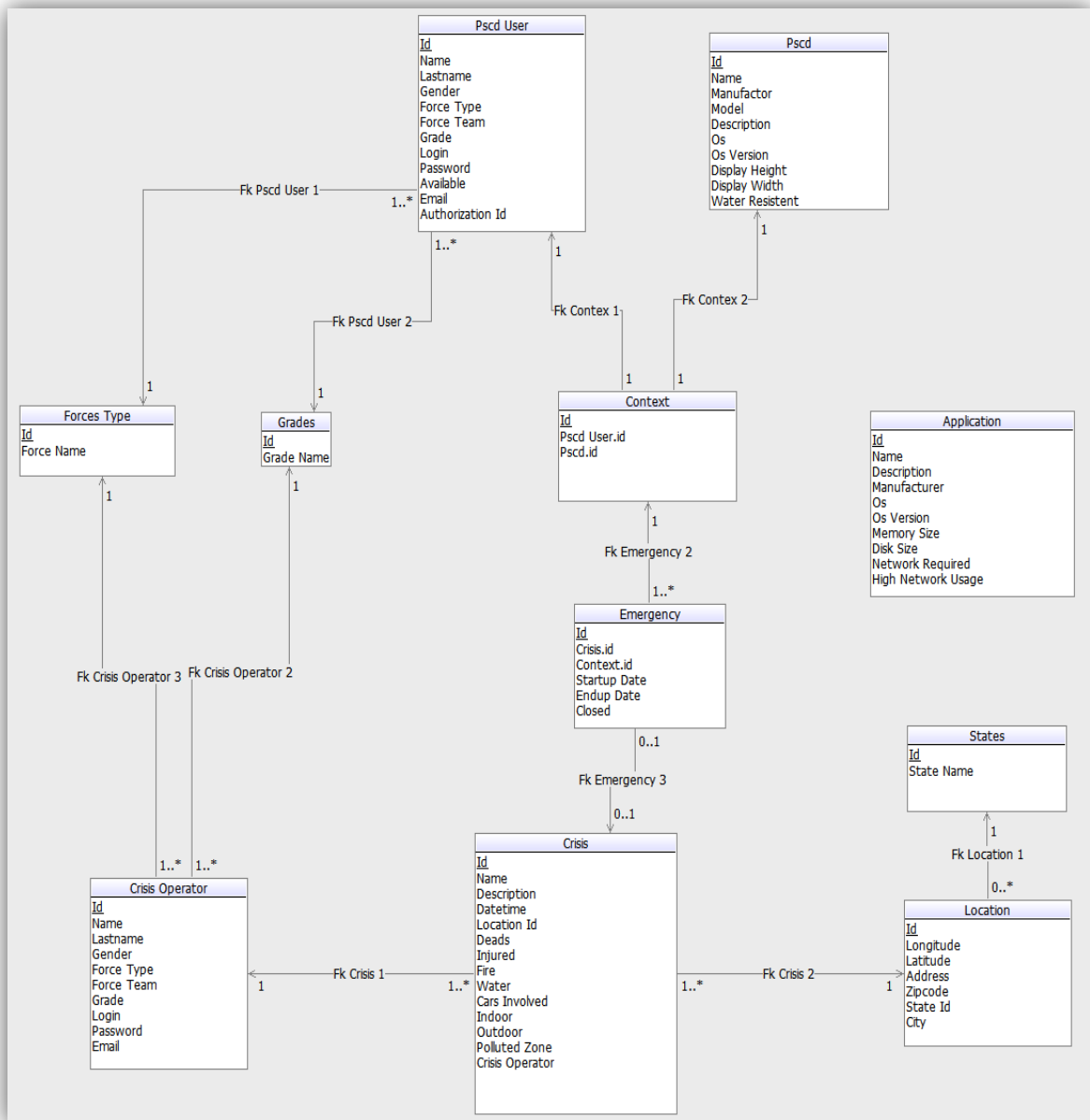
In PICO architecture there is a difference between persistent and dynamic data. On one side, we have to store some data which will be shared among all PSCD Users registered to the system. To do this, a relational database is used.

Examples of persistent data are:

- User: It is an entity which contains all users details necessary to perform authorization and authentication
- Devices: it provides all the features of the devices. It is necessary to discover if a specific user has sensor, the size of memory and so on.

On the other side, there are dynamic data such as temporal, geo-location and Environment that will not be stored but will be used by rule engine to perform reasoning.

A draft example of the E-R Diagram is show in figure Figure 1: E-R Model of PICO Server (PSCDS)



**Figure 1: E-R Model of PICO Server (PSCDS)**



---

## 3 Prototype Design

---

### 3.1 PICO Server (PSCDS)

---

#### 3.1.1 Technologies

##### 3.1.1.1 Java

PICO Server is written in Java. The Java version used in PICO is the 1.6. Java is a programming language expressly designed for use in the distributed environment of the Internet. It was designed to have the "look and feel" of the C++ language, but it is simpler to use than C++ and enforces an object-oriented programming model.

##### 3.1.1.2 Tomcat

As application container, PICO Server uses Apache Tomcat. Tomcat is a servlet container that is used in the official Reference Implementation for the Java Servlet and JavaServer Pages technologies. Apache Tomcat is developed in an open and participatory environment and released under the Apache Software License.

##### 3.1.1.3 PostgreSQL

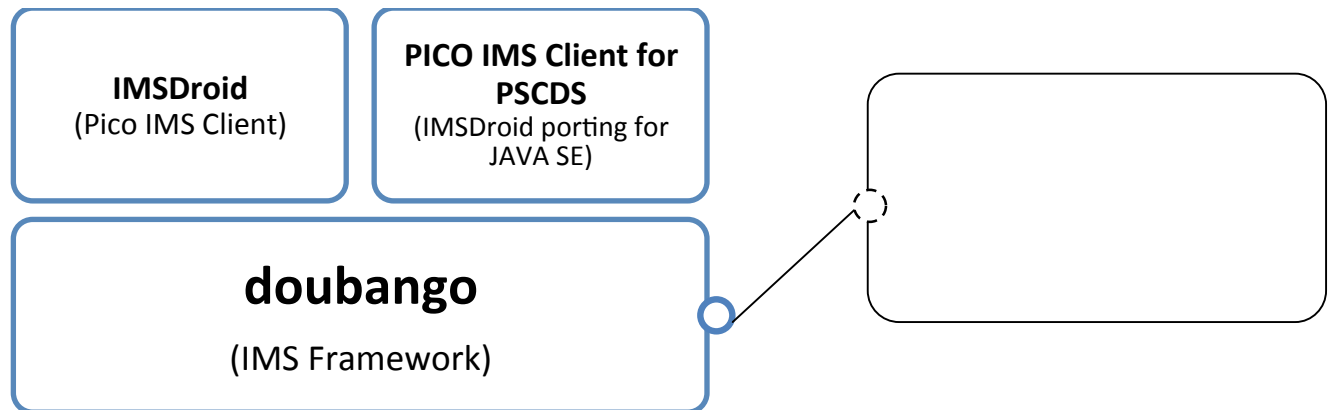
PICO server uses PostgreSQL as database. PostgreSQL is a powerful, open source relational database system. It has more than 15 years of active development and a proven architecture that has earned it a strong reputation for reliability, data integrity, and correctness. It runs on all major operating systems, including Linux, UNIX and Windows.

##### 3.1.1.4 Doubango IMS Porting

Doubango is an advanced open source 3GPP IMS/RCS framework for both embedded and desktop systems. The framework is written in ANSI-C to ease portability and has been carefully designed to efficiently work on embedded systems with limited memory and low computing power and to be extremely portable.

---

IMSDroid is an IMS client for Android system based on doubango framework. Unfortunately there was no server/desktop IMS client to use for PICO Server therefore, also in collaboration with the main developer of doubango, an IMS client based on the framework has been ported to run inside PICO Server

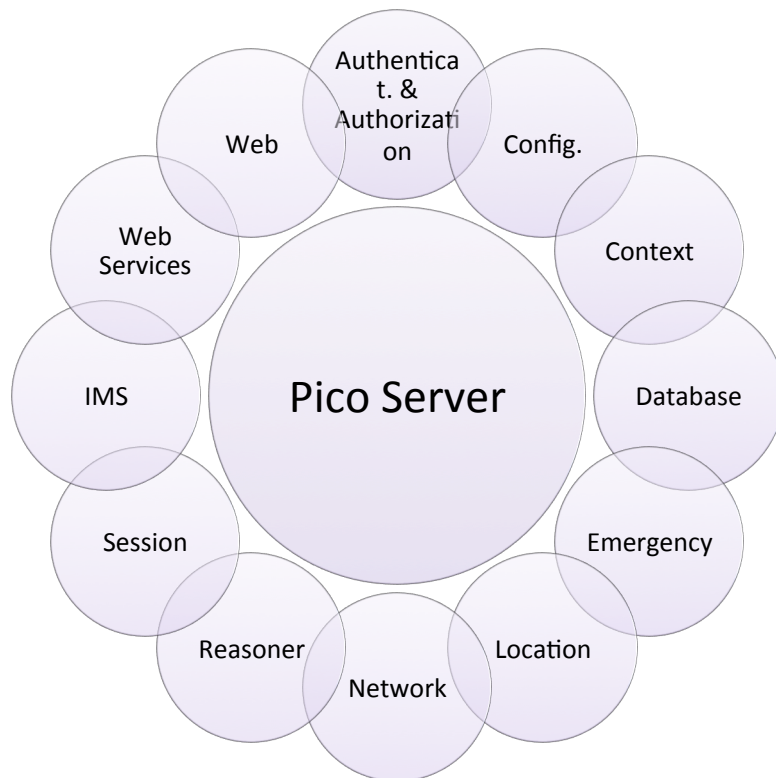


**Figure 2: We ported the IMSDroid (Android IMS Client) to Java Standard to include IMS features also in PICO Server. Both are based on doubango framework which is written in C Language.**

---

### 3.1.2 Subsystems

Pico Servers (PSCDS) is composed by several subsystems. Each of them offers a subset of features in the system.



**Figure 3: PICO Server subsystem**

---

---

### 3.1.3 Source code structure and packages

This is the final packages structure based on identified subsystem:

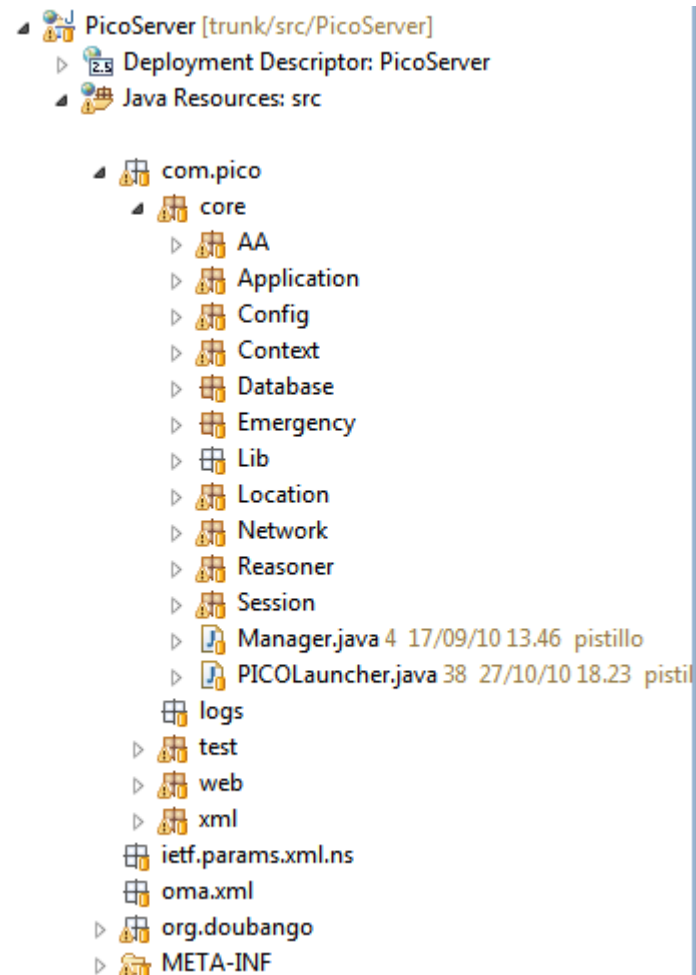


Figure 4: Final packages structure for PICO Server (PSCDS)

---

### 3.1.4 General Class Diagram

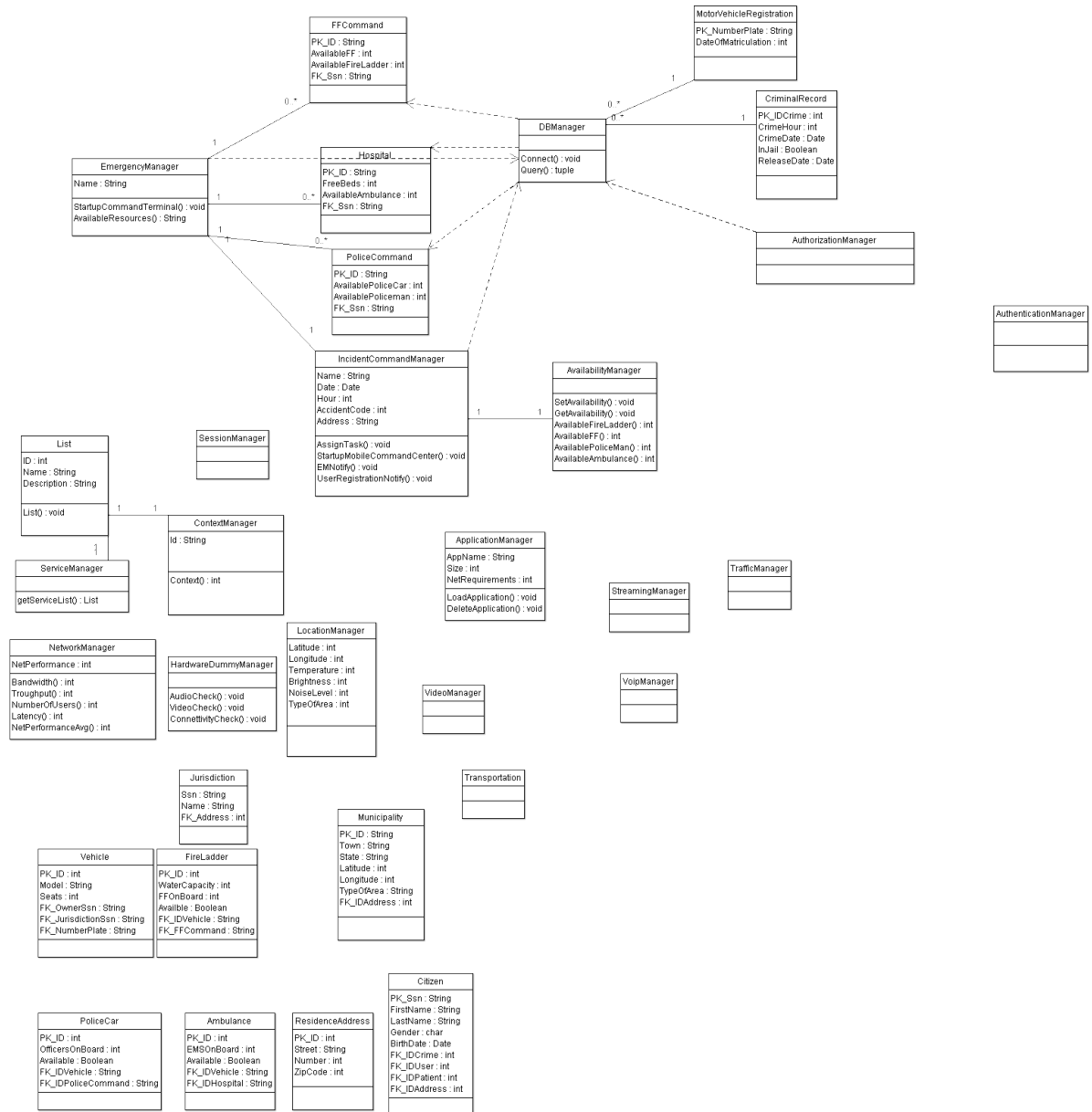


Figure 5: Classes overview for the PICO Server

---

## **3.2 PICO Client (PSCD)**

---

### **3.2.1 Technologies**

Technologies involved for the PICO Client are:

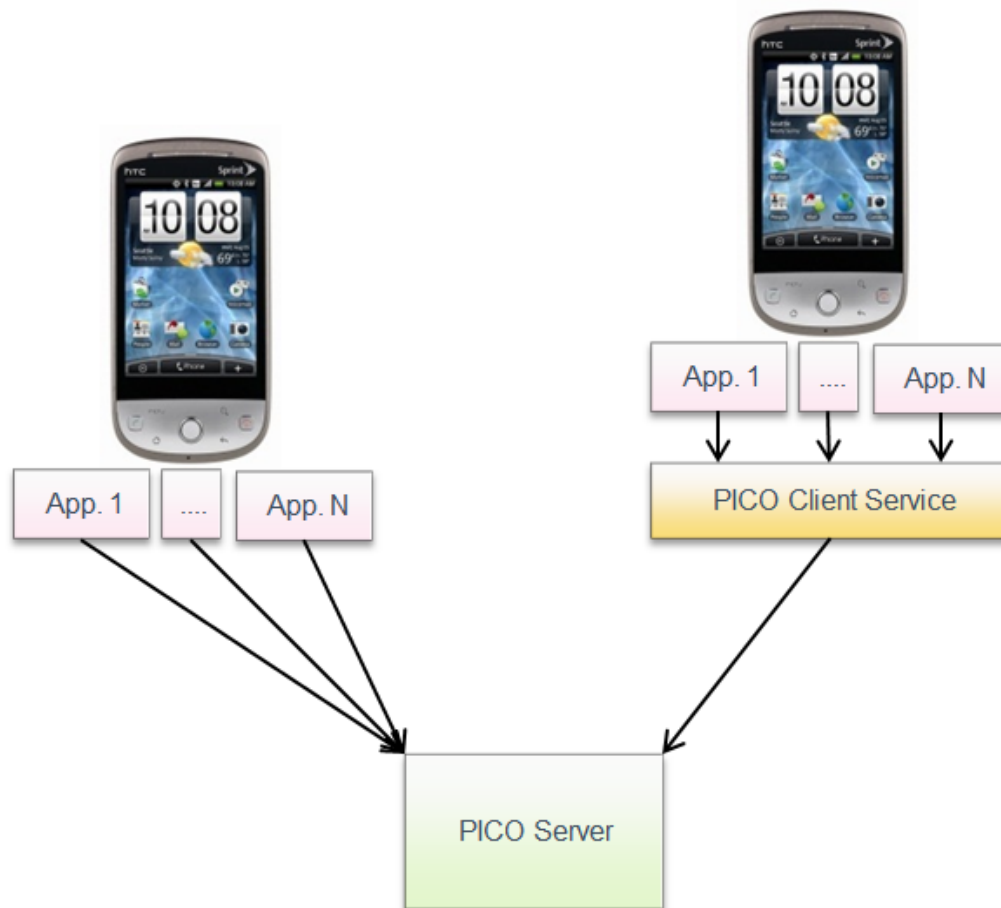
- IMSDroid as IMS client. (it has been extended)
- Android Framework (Services, Layout)
- ConteXML specific

---

### 3.2.2 Subsystems

The subsystems of PICO Client are:

- ContextManager
- Desktop Manager
- Service
- IMS Manager
- UI Manager



**Figure 6: The PICO Service manages the context updates received from the PICO Server. It manages also the applications that will be installed on the device**

---

---

### 3.2.3 Source code structure and packages

This is the final packages structure based on identified subsystem:

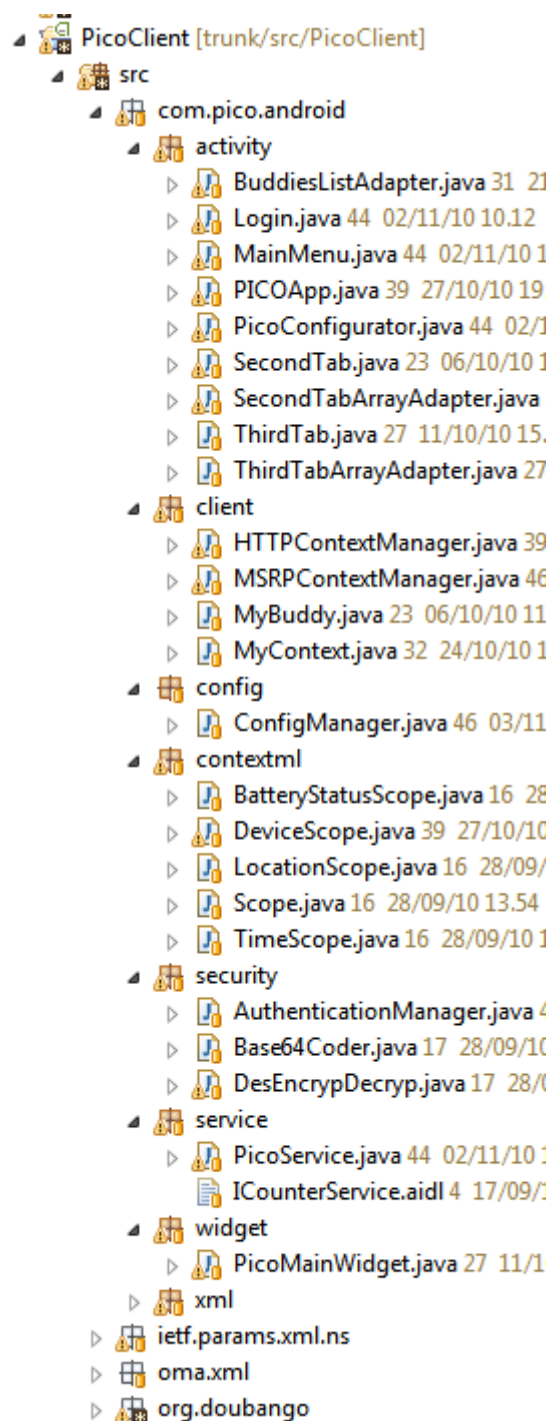


Figure 7: Final packages structure for PICO Client (PSCD)

---



---

## 4 Prototype Scenario

---

### 4.1 Storyboard

---

1. A car accident (A1) with three unconscious injured and 3 cars involved has just happened in the Zone 1 (Z1).
2. A policeman (PSU1) reaches the Accident (A1).
3. The main desktop of the policeman's device (PSCD1) offers the list of buddies and the position of the policeman in that moment. Among the list, the policeman selects the operator of the police station (OP1) and begins an audio call using a IMS Client.
4. During the call, the operator (OP1) inserts via a web interface all initial data about the accident into PICO Server (S1) and share the application "First Emergency Call" (AP1) with him.
5. In the notification area of the policeman's device the download status of the application (AP1) is shown, when it is downloaded, the policeman (PSU1) from the second desktop of the device (PSCD1) launches the first emergency call application (AP1) and starts scanning the QRCode of the first victim (V1)
6. When Qr code is scanned, the application (AP1) shows a pop-up message with the Name and Lastname of the victim (V1) and calls automatically the first number of an important person (eg. Parents) using the IMS Client. During the call the policeman warns the situation and read the name to the person.
7. At the same time, the application using the information embedded in the QRCode, sends the name and other important information related to the victim (V1) to the nearest PICO Server (PICO1) available ( using LoST , future development\*)

Moreover all Accident information such as name and current location are also sent to his important facebook contacts.\*\*

8. PICO Server processes the information related to the accident (A1) and the information related to all users' contexts connected to the PICO Server. The reasoner of the PICO Server (PICO1) begins to analyze contexts and positions closer to the accident(A1). Reasoner will select the best PSCUser available according also to other parameters such as: Team, kind of user....
  9. An ambulance (PSU2) with two paramedics is near the Zone 1(Z1), PICO Server(PICO1) via the reasoner alerts with a message the paramedic's device (PSCD2) that contains all medical informations gained at 7) , using the embedded IMS Client, automatically connects with an audio call or video call (base on network condition, battery level), the paramedic(PSU2) with the policeman (PSU1). On the main desktop is shown a map with 2 IMS buddies, PSU1 and PSU2 and the best path to reach the Accident (A1) base on traffic conditions.
  10. On display of the PSCD2 a message with the availability of a diagnostic application (AP2) with some custom data related to the victim (V1) is shown. The paramedic (PSU2) is able to download and install it during the route to the accident (A1).
-

- 
11. The policeman starts from the contextual application list(second desktop) an application to find the nearest tow truck to the zone 1 (Z2) and sends a sms with all accident details (location, number of cars )
  12. A squad of firefighetrs is in the Zone 2 (Z2) attached to PICO Server 2 (PICO2). The squas is available and using LoST, the request is forwarded from PICO1 to PICO2.
  13. PICO Server starts the application that shows the best path to the FF and begins also a communication session between the FightFigher (PSCU3) and the policeman (PSCU1) based on nework condition:
    - a. chat session (low traffic condition) (the first auto chat message contains all accident details)
    - b. Audio session (medium traffic condition)
    - c. Video session (high traffic condition)
  14. The policeman now has 3+1 active buddies in his buddy list and based on session started on 13), decides to:
    - a. share a picture using a IMS session (IMS Client) to the FF cause a small fire was started after the accident. So the FF can see how big is the fire and position of the cars.
    - b. ....
  15. The reasoner waiting the ambulance, updates the second desktop of the Policeman (PUS1) device based on his contex (battery level):
    - a. With a First Aid application (in case of medium level battery). (second desktop)
    - b. With a PDF file with the aid instructions to download (third desktop).
      - i. If the device has low disk space, then a PDF lite version will be downloaded.
      - ii. If the device has high disk space, then a PDF full version will be downloaded.

Moreover the reasoner can activate/deactivate some sensors of the PSCD Device (GPS, Camera, wifi, screen light...)

16. A notification message on the PSCD1 device aletes the policeman that the contextual applicaiton list (second desktop) has been updated, so the policeman can switch the desktop, selects the First Aid Application (or PDF file from third desktop) and download/launch it.
  17. When the ambulance is arrived to the Accident (A1), the paramedic selects from the contextual application(second desktop) list the application to monitor the heart rate of the Victim
  18. TODO: PICO Answer machine. Automatic text answers via IMS chat
-

---

## 5 User Interface

---

### 5.1 Introduction

---

The visual aspect is certainly important to provide access to all applications installed. In PICO the context desktop metaphor (or Tab application) has been used. A bit like what happens in the *HTC Sense*<sup>1</sup> interface, there are several desktops that fit and provide different information depending on context. With fairly large screens of recent device, each desktop can be split in more portions and each portion can offer to the user different kind of information. It is not anymore a place in which just the icons are in, but an interactive area with its own added value.

### 5.2 User interface and visual design

---

With PICO some guidelines have been defined for the user interface development. Essentially on the mobile application there are three main desktop, or three main application tabs<sup>2</sup>.

- a. Main Desktop (Tab), see Fig. 1: Is the main desktop for the PSCD which contains all useful information related to the emergency. It consist of:
  - Status Info Widget/Box: Emergency details
  - Buddies List (Operator + other buddies associated to the emergency)
  - Map: Location of all PSCD Users and/or best path available to reach the Emergency.
- b. Second Desktop (Tab): (Contextual application list)
  - It will contain all Contextual applications (installed or to be installed)
  - It will contain either previously installed applications or not installed applications which will be links to allow the download of them
- c. Content Desktop(Tab) (Content):
  - Content file List: It contains useful contents to the PCSD User such as PDF, Images etc.

---

<sup>1</sup> **HTC Sense:** is a user interface deployed by HTC to extend Android UI.

<sup>2</sup> **Application Tabs:** are application sections

---

Moreover all notifications to the PSCD users will be delivered through a notification area on top of the main widget using SIP/IMS messages.

PICO



**Fig. 1**

---

## Index of Figures

---

Figure 1: E-R Model of PICO Server (PSCDS) .....	8
Figure 2: We ported the IMSDroid (Android IMS Client) to Java Standard to include IMS features also in PICO Server. Both are based on doubango framework which is written in C Language. ....	10
Figure 3: PICO Server subsystem.....	11
Figure 4: Final packages structure for PICO Server (PSCDS).....	12
Figure 5: Classes overview for the PICO Server.....	13
Figure 6: The PICO Service manages the context updates received from the PICO Server. It manages also the applications that will be installed on the device .....	15
Figure 7: Final packages structure for PICO Client (PSCD).....	16

---