

POLICLINICO2NA PATIENTID

System Architecture Description

Versione 02

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

Emissione

	Nome e Cognome Ruolo - Ente	Firma	Data Firma
Redatto da	Ing. Gianrico D'Angelis Progettista - KIRANET' srl		
Verificato da	Ing. Raffaele Chianese Dir.Tecnico - KIRANET' srl		
Approvato da	Ing. Raffaele Chianese Dir.Generale - KIRANET' srl		

N.B.: Questa è una copia autorizzata solo se è firmata o siglata con inchiostro non nero.

Versione

Versione	Data	Descrizione - Motivazione	Autore
00	18/10/11	Bozza	Gianrico D'Angelis
01	16/11/11	Prima emissione	Gianrico D'Angelis
02	01/12/11	Seconda Emissione	Gianrico D'Angelis
03			
04			

Proprietà

Tutti i diritti del presente documento sono riservati. © Copyright 2005-2011. KIRANET S.r.l.

Il presente documento ed i suoi contenuti sono proprietà di KIRANET s.r.l.. Altri diritti relativamente al documento sono determinati dalle disposizioni applicabili di legge. Il presente documento è stato fornito alle seguenti condizioni: nessun diritto o nessuna licenza relativi al presente documento o ai suoi contenuti sono concessi o derogati con la fornitura del presente documento. Il presente documento o i suoi contenuti non devono essere usati o trattati in alcun modo contrario ai diritti di KIRANET s.r.l. o contro i suoi interessi e non devono essere resi noti ad altri senza previo consenso scritto da parte di KIRANET s.r.l..

Tutti i marchi citati sono dei legittimi rispettivi proprietari.

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

Indice

1. INTRODUZIONE	5
1.1 SCOPO DEL DOCUMENTO	5
1.2 DEFINIZIONI, ACRONIMI E ABBREVIAZIONI	5
1.3 DOCUMENTI DI RIFERIMENTO	5
1.4 PANORAMICA	6
2. VINCOLI E OBIETTIVI ARCHITETTURALI	7
3. MODELLO DEI CASI D'USO	8
4. RAPPRESENTAZIONE LOGICA	9
4.1 ARCHITETTURA GENERALE	9
4.2 SOTTOSISTEMA "SOFTWARE SCRITTURA"	9
4.2.1 PATIENTIDWRITER	10
4.2.2 WRITECONTROLLER	10
4.2.3 WEBSERVICE	10
4.2.4 USERINTERFACE	11
4.2.5 MAINWINDOW	11
4.2.6 DBCONNECTOR	12
4.2.7 UTIL	12
4.2.8 WRITERRFIDAPI	12
4.2.9 PACKAGE EP-ISO15693.DLL	13
4.3 SOTTOSISTEMA "SOFTWARE LETTURA"	13
4.4 SOTTOSISTEMA "DBSERVER"	14
5. RAPPRESENTAZIONE DEI PROCESSI	15
5.1 SEQUENCE DIAGRAM	15
6. SCHEMA DI DEPLOYMENT	16
7. IMPLEMENTAZIONE E SVILUPPO	17
7.1 STRUTTURA GENERALE	17
7.2 DETTAGLI DI IMPLEMENTAZIONE	17
7.2.1 FILE DI CONFIGURAZIONE	17
7.2.2 ALGORITMO DI FIRMA	18
7.2.3 WINDOWLESSRFIDDEFAULTHANDLERIMPL.CPP	18

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

8. DIAGRAMMA DEI DATI **19**

9. DIMENSIONAMENTO E PERFORMANCE **20**



Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

1. Introduzione

1.1 Scopo del documento

Il presente documento ha lo scopo di definire l'architettura del sistema relativo al progetto POLICLINICO2NA PATIENTID, realizzato da KIRANET s.r.l. (spesso per semplicità anche solo "KIRANET") per AOU Federico II di Napoli (di seguito "il committente").

1.2 Definizioni, acronimi e abbreviazioni

Di seguito sono riportati tutti i termini, gli acronimi e le abbreviazioni necessarie per interpretare correttamente il documento.

Definizioni	
Ospedalizzazione	Nel contesto di riferimento, un paziente è definito "ospedalizzato", nel momento in cui si ricovera e viene aperta una cartella clinica con i suoi dati.
Tag RFID	Con questo termine si indica, nel campo degli RFID, un'etichetta composta da un chip ed un'antenna.
Lettore RFID	E' una macchina dotata di hardware e software dedicati all'interazione con Tag RFID secondo uno standard protocollare scelto.
Toughbook	Computer rugged Touch Screen portatile allineone per settore sanitario con lettore RFID e codice a barre integrato
Codice Nosologico	Equivale al Codice di Cartella Clinica

Acronimi	
RFID	Radio Frequency Identification – Identificazione a Radio Frequenza.

Abbreviazioni	
CC	Cartella Clinica
CN	Codice Nosologico

1.3 Documenti di Riferimento

Interni:

- 00684.007.01 → POLICLINICO2NA PATIENTID – System Architecture Description – v.01
- 00684.003.03 → POLICLINICO2NA PATIENTID – System Requirements Specification – v.03

Esterni:

- IEEE/ANSI 830-1993 "IEEE Recommended Practice for Software Requirements Specifications"
- FDIS 12207, Systems and software engineering - Software life cycle processes
- Using a Single Business Pattern with the Rational Unified Process (RUP), IBM Redbooks Paper
- From Waterfall to Iterative Development – A Challenging Transition for Project Managers
- IBM Rational Unified Process, <http://www-128.ibm.com/developerworks/rational>
- Applying Requirements Management with Use Cases, IBM Rational

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

1.4 Panoramica

Nella prossima sezione sono descritti i vincoli e gli obiettivi dell'architettura del sistema. Nella terza sezione si trova il modello dei casi d'uso, mentre in quella successiva la rappresentazione logica a sottosistemi/template/classi. Nella quinta sezione è riportata la rappresentazione dei processi, delle collaborazioni e delle sequenze e in quella successiva lo schema di deployment. Nelle ultime due sezioni vi sono rispettivamente gli schemi a layer per l'implementazione e lo sviluppo e il diagramma dei dati.



Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

2. Vincoli e obiettivi architetturali

Questa sezione descrive l'architettura del sistema e come è rappresentata, enumera le “viste” necessarie utilizzate per il progetto e i diagrammi in esse contenuti.

Inoltre di seguito sono riportati i requisiti e gli obiettivi che hanno un impatto significativo sull'architettura (ad es. sicurezza, privacy, uso di prodotti commerciali, portabilità, distribuzione e riuso).

Infine si stabiliscono i vincoli speciali che devono essere applicati: strategie particolari di progettazione e di implementazione, strumenti di sviluppo da usare, eventuali prodotti legacy.

Il sistema relativo al progetto POLICLINICO2NA PATIENTID nasce allo scopo di realizzare l'identificazione del paziente “ospedalizzato”, in maniera automatica ed informatizzata, tramite l'impiego di tecnologie software ed hardware opportunamente scelte.

Nello specifico il sistema prevede l'impiego della tecnologia RFID con tag posizionato all'interno di un braccialetto da far indossare al paziente in fase di accettazione. Un software di codifica, installato nelle postazioni di accettazione, provvederà a codificare il braccialetto in modo univoco così da associarlo alla rispettiva cartella clinica del paziente. Un ulteriore software di lettura, installato su Tablet per applicazioni ospedaliere, provvederà a leggere il tag su braccialetto al fine di identificare il paziente per diversi scopi.

Il corretto funzionamento del sistema è subordinato all'esistenza di un'infrastruttura di rete Ethernet e WiFi correttamente configurate al fine di consentire la comunicazione tra quelle parti del sistema che ne richiedono l'uso.

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

3. Modello dei casi d'uso

Questa sezione elenca i casi d'uso e gli scenari.

Cod	Caso d'uso	Requisito di riferimento
01	Associa Cartella Clinica a tag RFID (PatientID)	FR01,FR04
02	Legge tag sul braccialetto	FR03
03	Indossa braccialetto	FR02

Gli attori del sistema sono i seguenti:

- *Il personale ospedaliero*, che:
 - dopo aver creato la cartella clinica del paziente, avvia la procedura di codifica del braccialetto RFID;
 - tramite lettore RFID integrato nel toughbook, identifica il paziente per diversi scopi.
- *il paziente*, che nel momento in cui diventa ospedalizzato, riceve e deve indossare un braccialetto dotato del tag RFID con il proprio identificativo;
- *il medico* che tramite lettore RFID integrato nel toughbook, identifica il paziente per diversi scopi

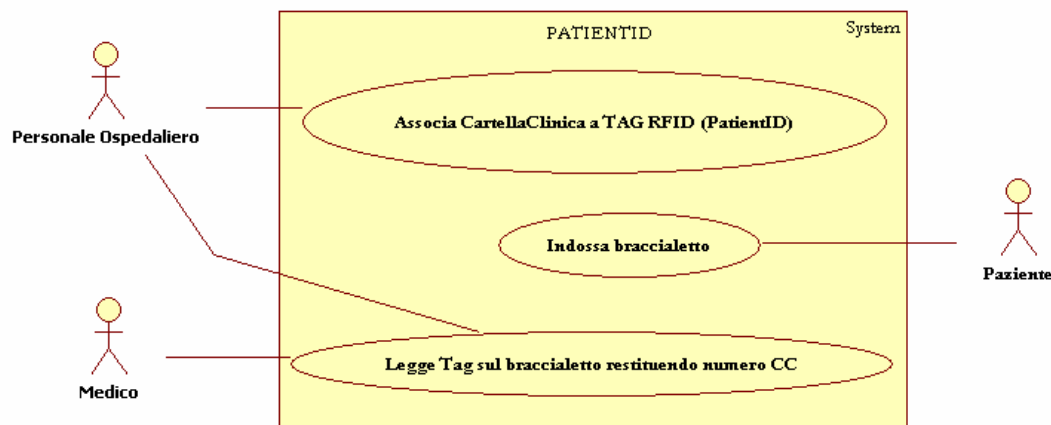


Figura 1 - Casi D'Uso 1

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

4. Rappresentazione logica

Questa sezione descrive l'architettura del sistema in generale, la sua eventuale decomposizione in sottosistemi e package e per ognuno di essi la decomposizione in classi, descrivendone le responsabilità, le relazioni, le operazioni e gli attributi.

4.1 Architettura generale

Il sistema relativo al progetto POLICLINICO2NA PATIENTID è suddiviso in tre Sottosistemi:

- Software Lettura;
- Software Scrittura;
- DBServer;

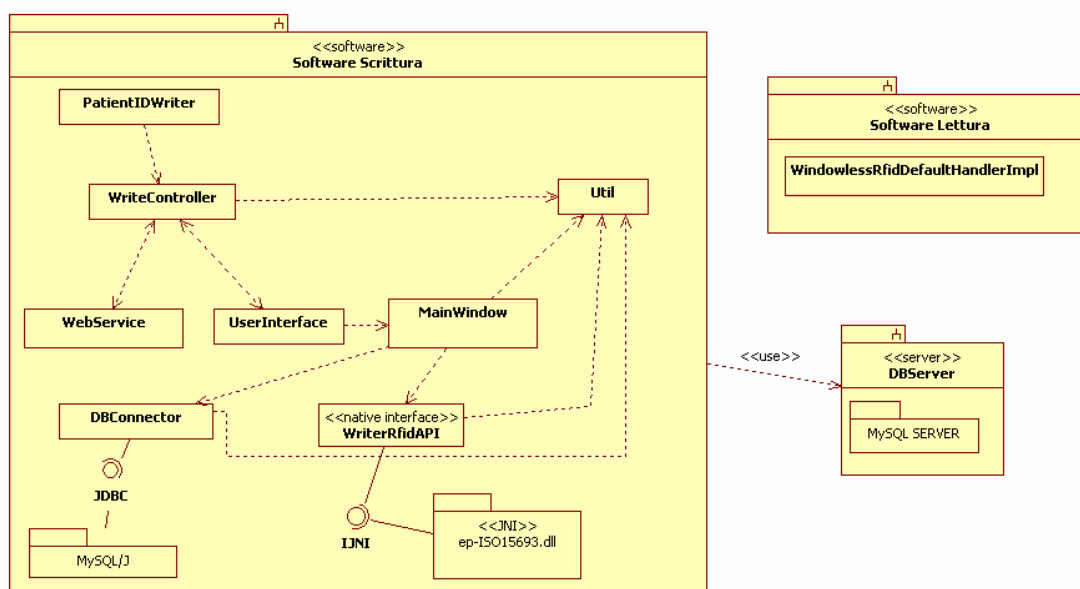


Figura 2 - Architettura Generale

Qui di seguito per ogni sottosistema si riporta una breve descrizione, il diagramma delle classi e i package contenuti; per ogni package si riporta una breve descrizione del diagramma delle classi; per ogni classe si riporta una breve descrizione e le responsabilità, le operazioni, gli attributi.

4.2 Sottosistema “Software Scrittura”

Il sottosistema Software di Scrittura si occupa della “Scrittura” dei braccialetti RFID. Per “Scrittura” intendiamo il processo di codifica del tag RFID e relativa associazione Tag-Nosologico su DBServer. Le classi e package componenti il software di scrittura sono:

- Classi
 - PatientIDWriter
 - WriteController
 - WebService
 - UserInterface

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

- MainWindow
- DBConnector
- Util
- WriterRfidAPI
- Packages
 - MySQL/J
 - ep-iso15693.dll

Di seguito è riportata una descrizione più approfondita per ogni classe.

4.2.1 PatientIDWriter

Questa classe contiene il methodo statico “main” e si occupa solo di instanziare un oggetto di tipo WriteController.

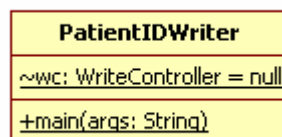


Figura 3 -- Classe PatientIDWriter

4.2.2 WriteController

Questa classe si occupa di avviare i due threads principali, uno per la gestione della User Interface e l'altro per la gestione del WebService. La classe usa metodi statici della classe Util per recuperare parametri di configurazione.

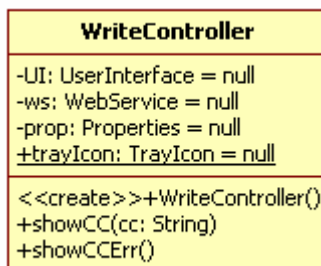


Figura 4 -- Classe WriteController

4.2.3 WebService

Gestisce il servizio il WebService in ascolto su localhost. Riceve ed interpreta la comunicazione HTTP. Instanzia un'oggetto WriteController per gestire le post-comunicazione.

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

```

WebService

-RES 200: String = "HTTP/1.1 200 OK\r\n"
-RES 202: String = "HTTP/1.1 200 OK\r\n"
-RES 400: String = "HTTP/1.1 400 Bad Request\r\n"
-RES 404: String = "HTTP/1.1 404 Not Found\r\n"
-RES HEADER_SERVER: String = "Server: KiraHTTserv 1.0\r\n"
-RES HEADER_CONTENTTYPE: String = "Content-type: text/html\r\n"
-COMM_PUTCC: String = "putcc"
-COMM_STATUS: String = "status"
-javascript: String = "<html><head><title>Kiranet XHR</title></head><body onLoad=self.close();</body></html>"
-port: int = 80
-wc: WriteController = null

<<create>>+WebService(port: int, wc: WriteController)
+run()

```

Figura 5 -- Classe WebService

4.2.4 *UserInterface*

Gestisce il posizionamento dell'icona nella systray, con relative creazione del menu contestuale. Instanzia la MainWindow.

```

UserInterface

-COMM1: int = 1
-COMM2: int = 2
+trayIcon: TrayIcon = null
-wc: WriteController = null
-command: int = -1
-message: String = ""
~newComm: boolean = false

<<create>>~UserInterface(wc: WriteController)
+createAndShowGUI()
#createImage(path: String, description: String): Image
+setCC(msg: String)
+setCCErr()
+dispatchCommands()
+run()

```

Figura 6 -- Classe UserInterface

4.2.5 *MainWindow*

Disegna la MainWindow e definisce gli handlers associati agli eventi. Instanzia oggetti di tipo WriterRFIDAPI e DBConnector.

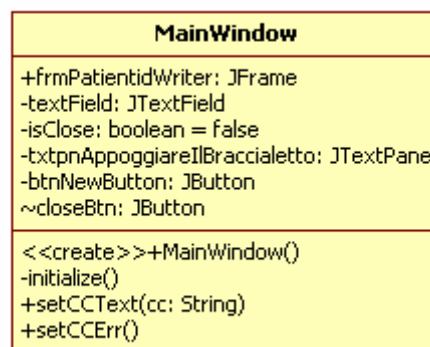


Figura 7 -- Classe MainWindow

4.2.6 DBConnector

Gestisce l'interfacciamento con il database.

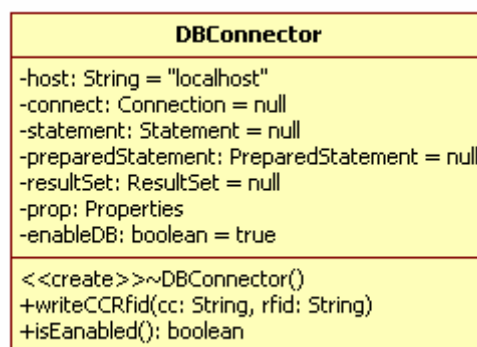


Figura 8 -- Classe DBConnector

4.2.7 Util

Contiene metodi statici di utilità generale.

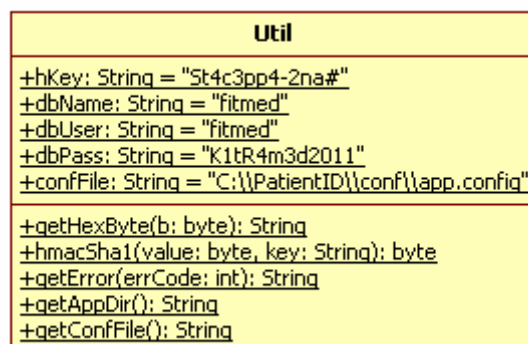


Figura 9 -- Classe Util

4.2.8 WriterRfidAPI

Interfaccia verso metodi Nativi.

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

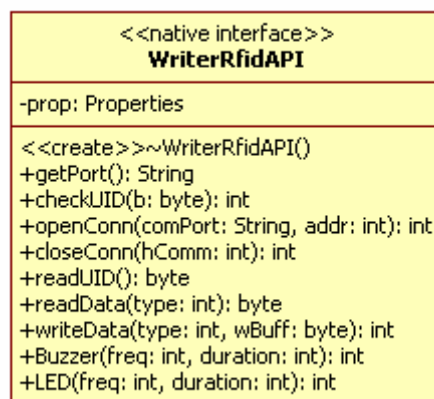


Figura 10 -- Classe WriterRfidAPI

4.2.9 Package ep-iso15693.dll

Questo package implementa i metodi nativi al fine esporre le API del lettore RFID (eprfid.dll) come metodi Java.

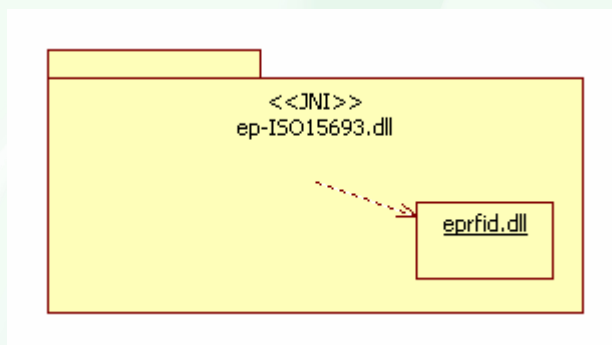


Figura 11 -- Pacjage ep-iso15693.dll

4.3 Sottosistema “Software Lettura”

Il sottosistema Software di Lettura si occupa della “Lettura” dei braccialetti RFID. Per “Lettura” intendiamo il processo di lettura fisica del tag RFID e relativa lettura del corrispondente Codice Nosologico. Il Package e la classe componente il software di lettura è:

- WindowlessRfidDefaultHandlerImpl

Il metodo GrabData realizza la lettura del campo dati nel tag RFID

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

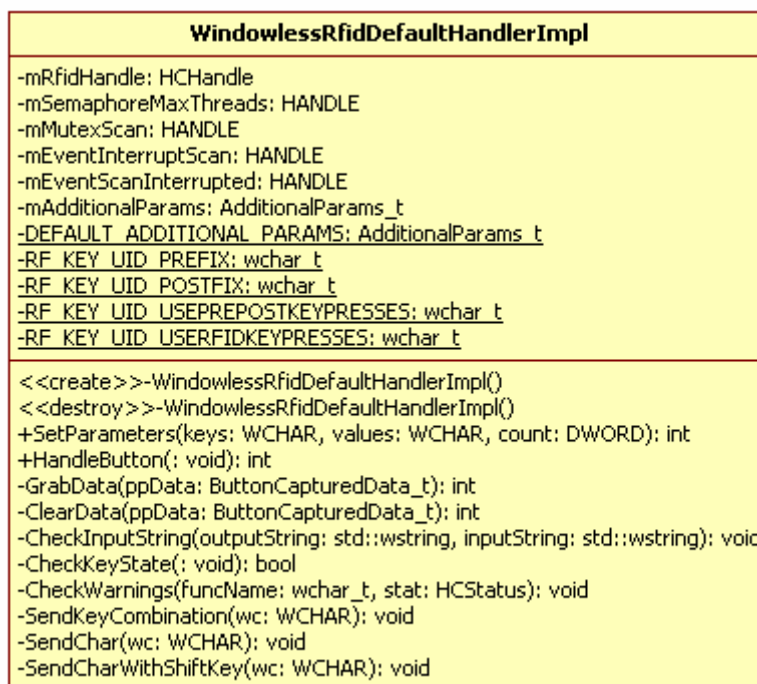


Figura 12 -- Classe WindowlessRfidDefaultHandlerImpl

4.4 Sottosistema “DBServer”

Il sottosistema DBServer sarà composto da un server remoto sul quale è installato il database server MySQL contenente la base dati con le necessarie tabelle per registrare le associazioni Codice Nosologico/tag RFID.

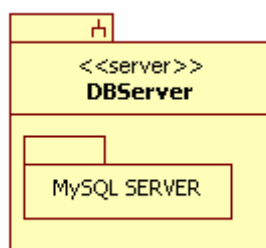


Figura 13 -- DBServer

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

5. Rappresentazione dei processi

Questa sezione descrive la decomposizione del sistema in processi leggeri (singoli thread di controllo) e processi pesanti (raggruppamenti di processi leggeri), organizzati in maniera da riunire quelli che comunicano e interagiscono, e descrive le tecniche con cui avviene la comunicazione (message passing, interrupts o rendezvous).

Processi del sistema:

- Codifica Braccialetto
- Lettura Braccialetto

Il processo di lettura si suddivide nei seguenti Threads:

- Threads del processo Codifica braccialetto
 - Leggi Nosologico
 - Lancia GUI
 - Scrivi RFID

5.1 Sequence Diagram

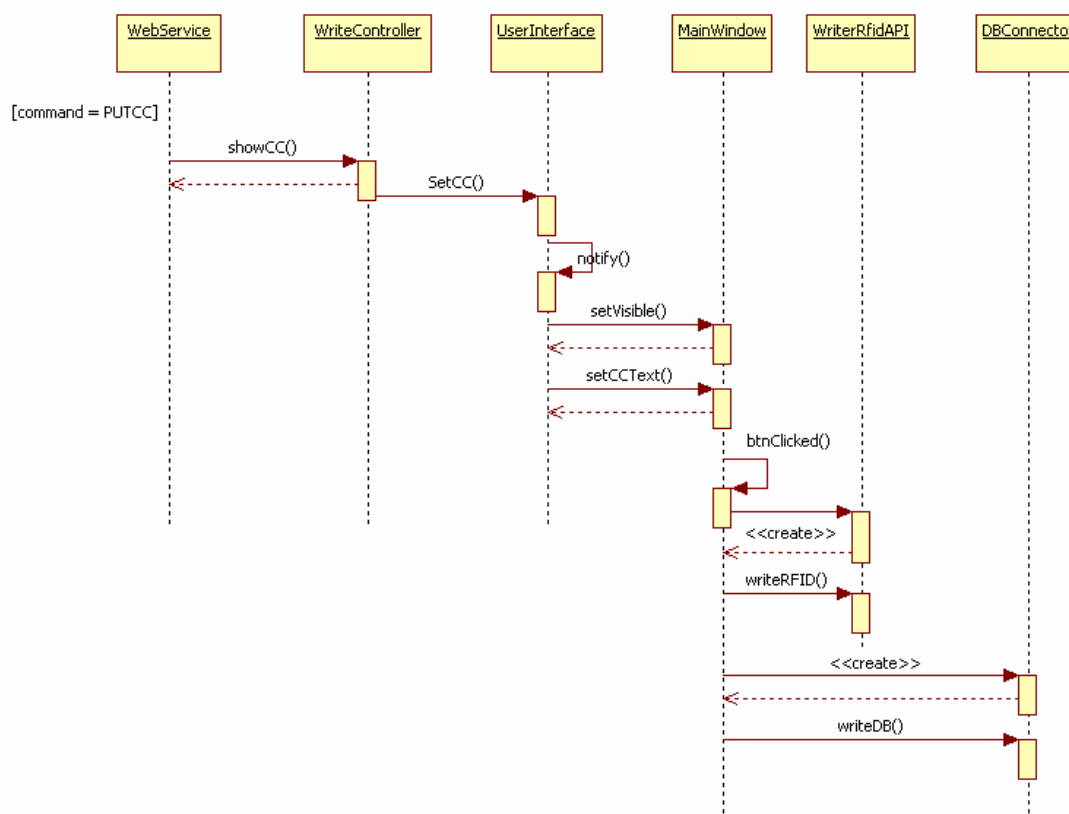


Figura 14 -- Scrivi Braccialetto

Per la lettura si faccia riferimento ai dettagli implementativi.

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

6. Schema di deployment

Questa sezione descrive le configurazioni delle reti e dell'hardware. Per ogni configurazione sono indicati i nodi fisici (server, workstation, ecc.) e le loro interconnessioni (bus, LAN, point-to-point, ecc.).

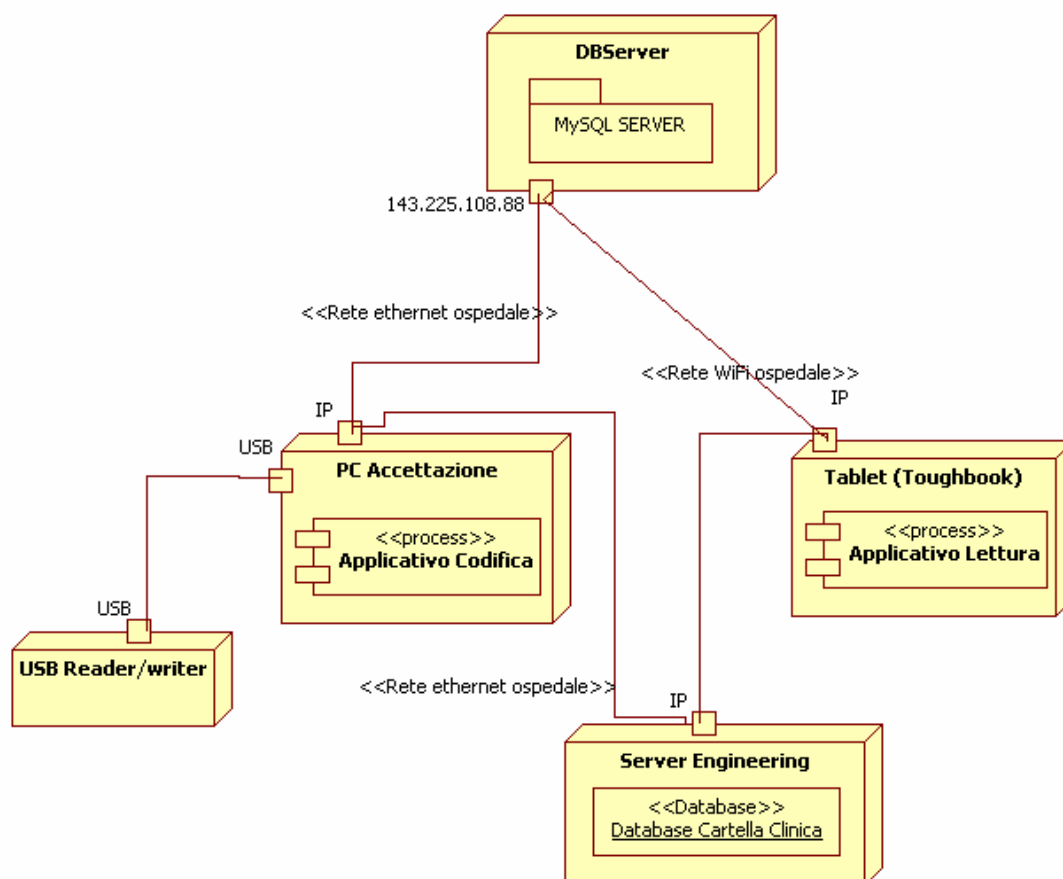


Figura 15 - Deployment Diagram

7. Implementazione e sviluppo

Questa sezione descrive la struttura generale del modello di implementazione, la decomposizione del sistema in layer e sottosistemi.

7.1 Struttura generale

Si riporta un *Component Diagram* che mostra le relazioni tra i layer (ognuno con la relativa descrizione, le regole che governano l'eventuale inclusione a un altro layer e le interfacce con gli altri layer).

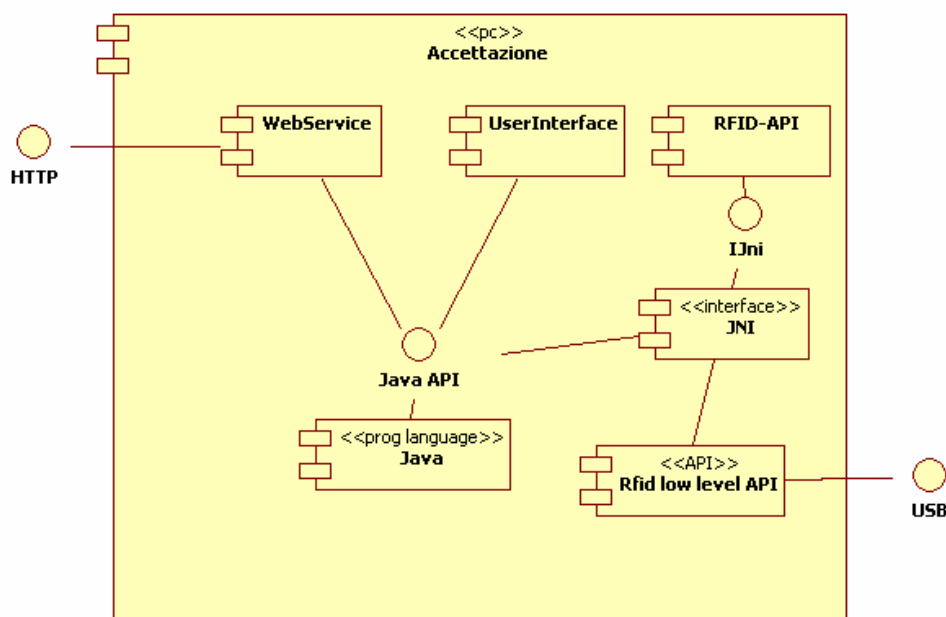


Figura 16 - Component diagram - scrittura

Per la lettura si faccia riferimento ai dettagli implementativi riportati di seguito.

7.2 Dettagli di implementazione

Si riportano alcuni dettagli e convenzioni per l'implementazione delle funzionalità.

7.2.1 File di configurazione

File di configurazione: %homepath%\PatientID\conf\app.config	
Nome Parametro	Valore
port	<serialport>
localport	local web service port, da non cambiare in produzione
dbhost	Database server nel formato host:port. Se impostato su none disabilita il database server

7.2.2 Algoritmo di firma

L'algoritmo di firma è uno HMAC SHA1 su una versione elaborata dell' UID del tag.
Password disponibile nella classe Util:

```
public static final String hKey = "St4c3pp4-2na#"
```

UID del tag elaborato nel seguente modo:

0xaa 0x00 0x0c 0x00 0x01 0x00 0x00 UIDREALE[7] ...UIDREALE[0] BCC 0xbb

BCC = 0x0c \oplus UIDREALE[7] \oplus ... \oplus UIDREALE[0] (\oplus is XOR)

7.2.3 WindowlessRfidDefaultHandlerImpl.cpp

E' il file che implementa la dll per il lettore. Versione modificata del file negli esempi nel SDK relativo ad intel MCA. Vedi codice allegato alla documentazione.

Da compilare come specificato nel manuale "Intel DHDK Developer Guide" nel SDK relativo ad intel MCA.

Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05

8. Diagramma dei dati

In questa sezione si riporta la descrizione della base dati persistente del sistema.

Il database di nome “fitmed” è composto da una tabella di nome “patientid” contenente i campi riportati di seguito:

id	Chiave primaria
CC	Codice Nosologico
RFID	UID Tag RFID
openDate	Data di registrazione
closeDate	Data di dimissione
active	Flag on/off, indica se l'associazione e' attiva o meno

```
CREATE TABLE IF NOT EXISTS `patientid` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `CC` varchar(15) NOT NULL,
  `RFID` varchar(35) NOT NULL,
  `openDate` datetime DEFAULT NULL,
  `closeDate` datetime DEFAULT NULL,
  `active` tinyint(1) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;
```

```
public static final String dbName = "fitmed";
public static final String dbUser = "fitmed";
public static final String dbPass = "K1tR4m3d2011";
```

9. Dimensionamento e performance

NA



Progetto: POLICLINICO2NA PATIENTID	Identificativo: 00684.007.02
Doc&Ver: System Architecture Description 02	Modulo: 00300.122.05