

tesi di laurea

# **Analisi e sviluppo di un client per l'accesso a dati su server remoto da dispositivi embedded**

Anno Accademico 2007-2008

relatore

**Ch.mo prof. Porfirio Tramontana**

correlatore

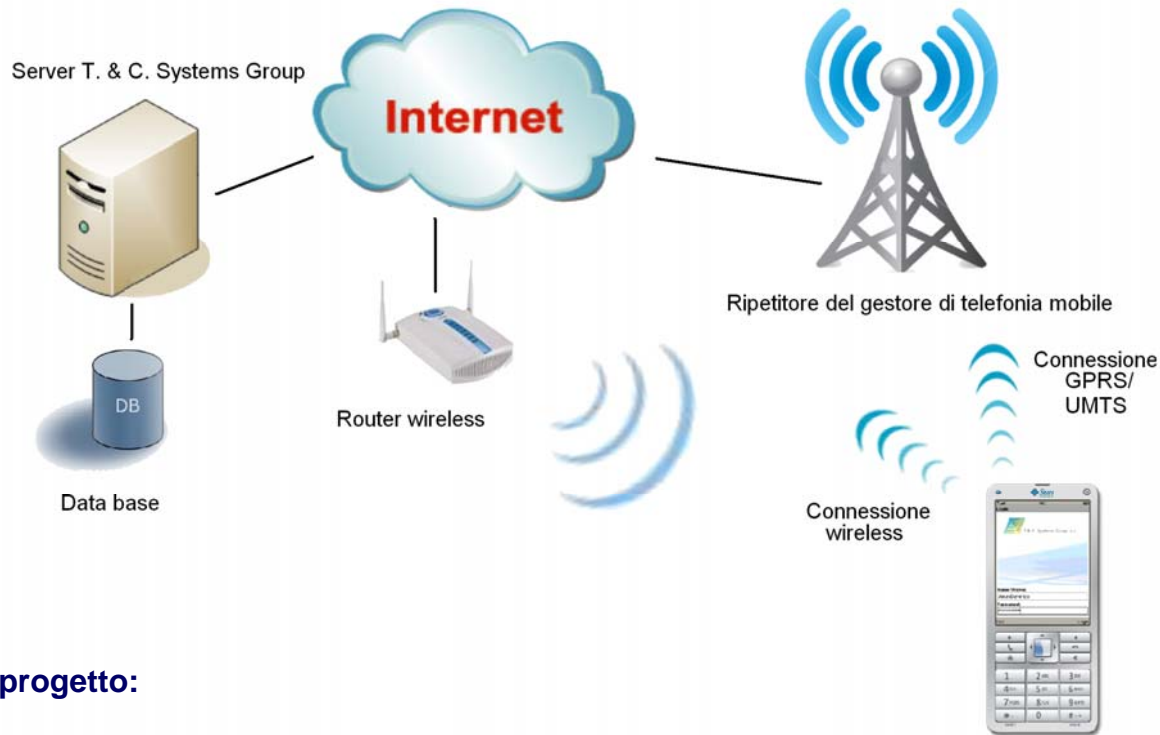
**Dott. Antonio Agliata**

candidato

**Domenico Amuro**

**Matr. 534/1239**





### Obiettivo del progetto:

- Client eseguibile su dispositivi embedded
- Sfruttando una connessione INTERNET (es. GPRS, UMTS o Wi-Fi) garantisce l'accesso ad informazioni aziendali presenti sul server centrale
- Massima compatibilità verso un maggior numero di dispositivi mobili



- **Necessità dell'azienda T.&C. Systems Group:**
  - **Comunicazione rapida e semplice con dipendenti fuorisede**
  - **Sviluppare un sistema integrato in un contesto più ampio di "Comunicazione Unificata"**

### Le comunicazioni oggi



- Sistemi di comunicazione verticali
- Comunicazione inefficiente
- Sovraccarico di comunicazioni
- Infrastrutture parallele

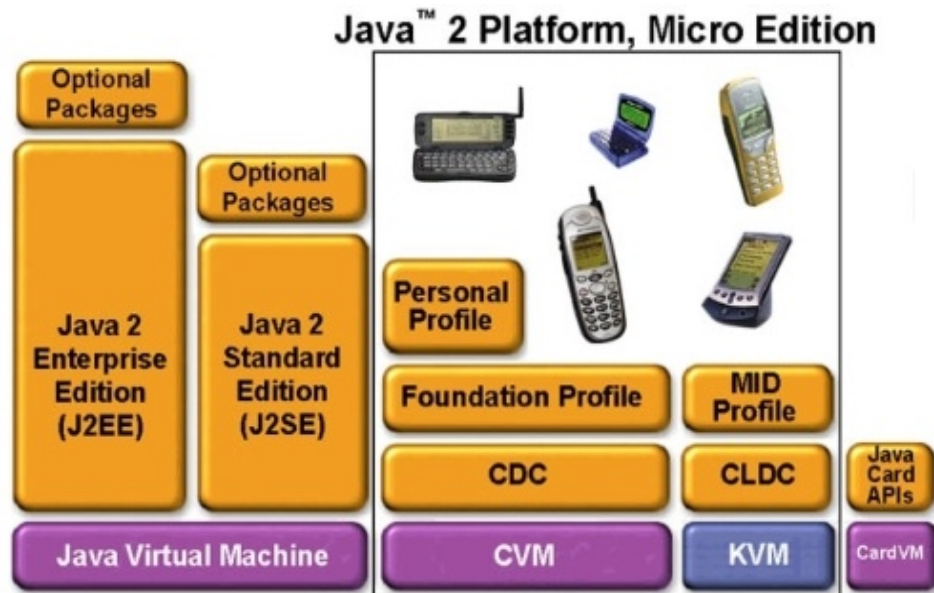
### Comunicazione Unificata



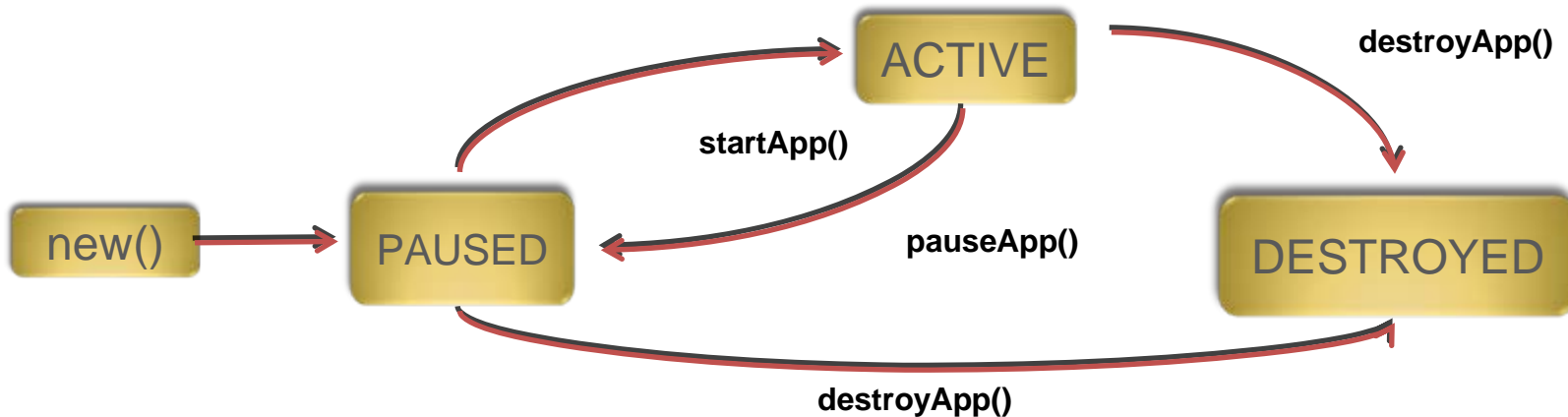
- Piattaforma comune integrata orizzontalmente
- Architettura distribuita e basata su standard
- Accessibilità nell'intera rete
- Un unico database comune



- **Java Micro Edition è la specializzazione di Java per lo sviluppo di applicazioni per telefoni cellulari, Mobile Devices**
  - Compatibilità con l'infrastruttura software preesistente (intranet aziendale)
  - Portabilità del linguaggio su un elevato numero di dispositivi
  - Semplicità nell'installazione dell'applicativo (JAR e JAD)
- **L'architettura di JME è suddivisa in 3 principali componenti:**
  - **K Virtual Machine:** Versione ridotta della JVM , funzionalità eliminate
  - **Connected Limited Device Configuration (CLDC 1.1)** :Classi fondamentali del Java
  - **Mobile Information Device Profile (MIDP 2.0)** :150 classi, Definisce il modello applicativo MIDlet



# La MIDlet



- Una MIDlet viene completamente gestita dall'Application Management System
- (AMS) che ne controlla l'installazione, l'avvio, la sospensione, il ripristino, la chiusura e l'eventuale rimozione.
- Queste applicazioni necessitano di una continua interazione con il sistema operativo del terminale, il quale deve poter sospendere o chiudere una applicazione in qualsiasi momento.
- *lazy initialization* (inizializzazione pigra)

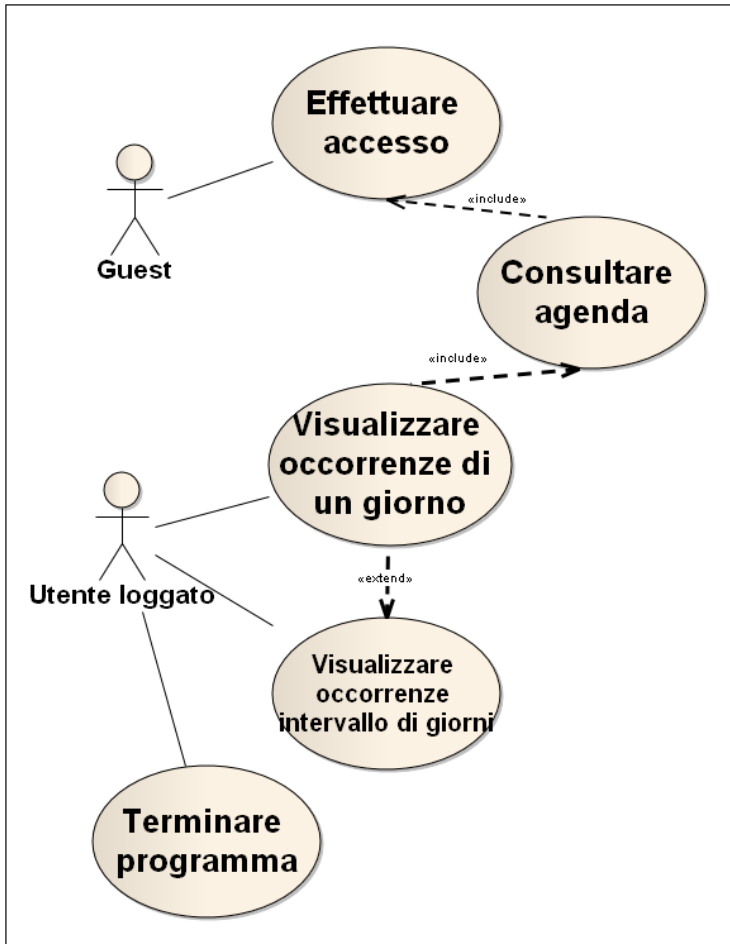


# Networking e parsing XML

- **Protocollo Hypertext Transfer Protocol (HTTP)**
  - Il più diffuso del web, usato anche per comunicazioni M2M
  - Sincrono, stateless, basato su request e response
- **HttpConnection ()**
  - `setRequestMethod()`, `setRequestProperty()` `openOutputStream()`
  - `getResponseCode()`, `getHeaderField()`, `openInputStream()`
- **eXtensible Markup Language (XML)**
  - Metalinguaggio di markup per la codifica di dati strutturati
  - È il più diffuso, garantisce alti livelli di interoperabilità tra sistemi, linguaggi di programmazione e S.O.
- **Librerie SAX 2.0 (Simple API for XML)**
  - Ottimizzate per dispositivi a basse prestazioni
  - Non mantiene in memoria il documento
  - Accesso sequenziale ai dati
  - Sistema di sola lettura del documento



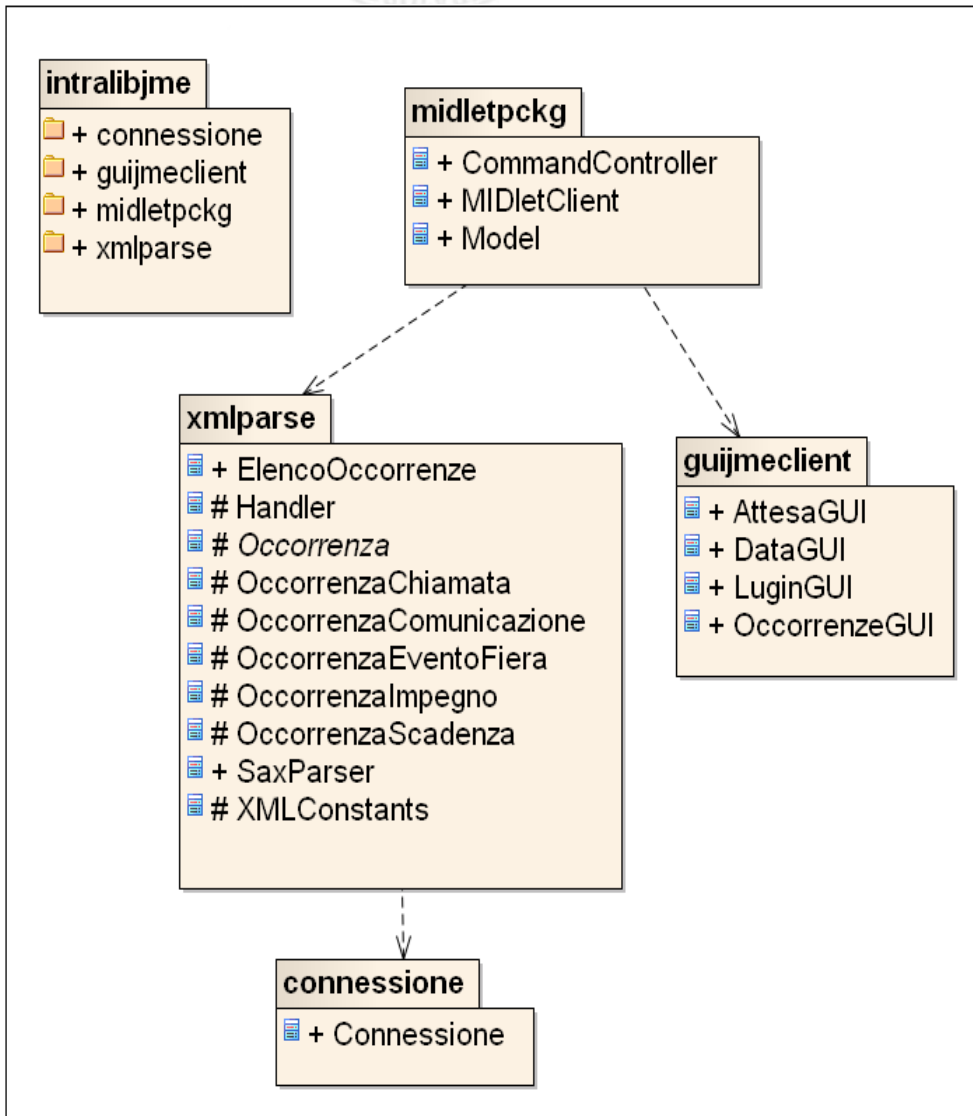
# Casi d'uso



- **Precondizione: accesso alla rete disponibile**
- **Effettuare accesso:**
  - Tramite l’inserimento di un nome utente e una password
- **Consultare agenda:**
  - Da un calendario visualizzato a video permette di selezionare una data di interesse
- **Visualizza occorrenza di un giorno:**
  - Scarica e mostra a video tutte le occorrenze relative ad una singola data
- **Visualizza occorrenza di un intervallo di giorno:**
  - Scarica e mostra a video tutte le occorrenze relative ad un intervallo di giorni



# System Design



## midletpkg

- MIDletClient l'entry point del programma
- CommandController, Modulo di controllo
- Model, mantiene traccia delle informazioni del sistema

## xmlparse

- SaxParser, classe principale per il parsing da file XML
- Handler, implementa la struttura popolata durante il parsing

## guijmeclient

- La GUI contenente tutte le forme utilizzate dall'applicativo

## connessione

- Modulo che implementa la connessione remota con protocollo HTTP

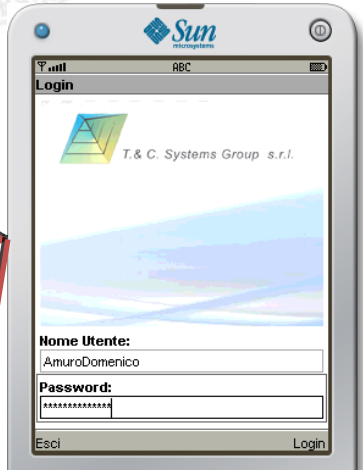




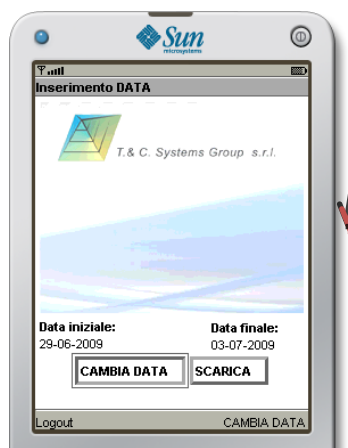
# Esempio d'uso



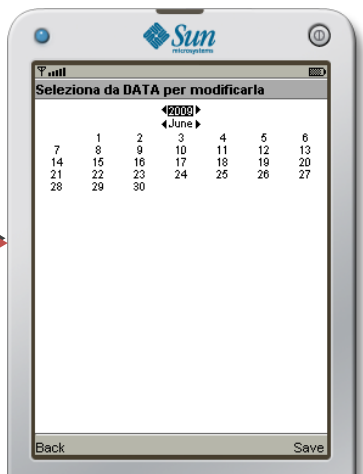
1- Lancio del software



2 - Fase di Login



3 - Cambio Data



4 - Selezione data



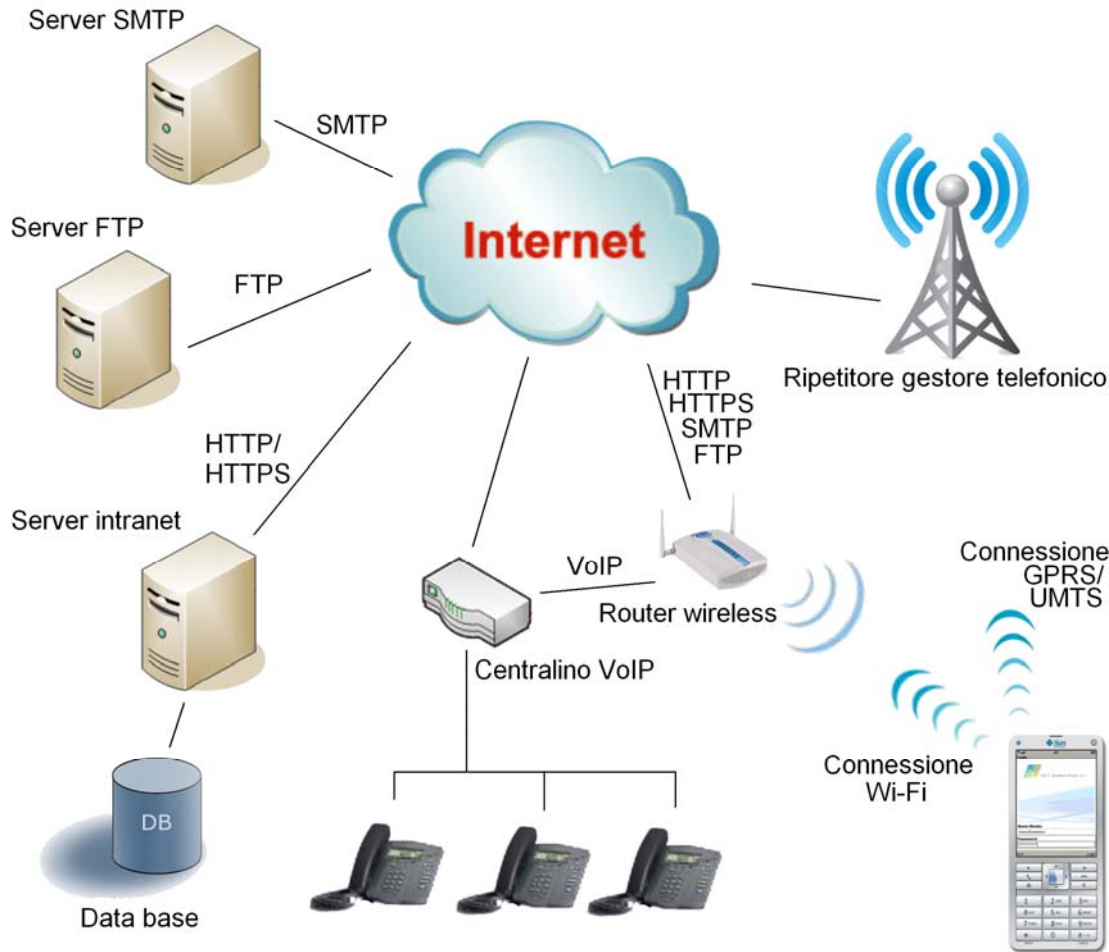
*Elenco a video delle occorrenze richieste*



# Conclusione

- **Primo passo verso un sistema di “Comunicazione unificata”**
- **Il prodotto sarà utilizzato :**
  - **da dipendenti fuori sede per brevi scambi di informazione limitati dalla connessione GPRS/UMTS**
  - **all'interno della rete aziendale sfruttando la più veloce connessione Wi-Fi, senza limiti di tempo**





# Sviluppi futuri

## Nuove funzionalità

- crea, modifica, cancella occorrenze

## Nuovi protocolli

- HTTPS
- FTP
- SMTP

## Tecnologia VoIP

