

Documento preliminare

Farisano Gino

Esposito Vincenzo

Scovotto Vanessa

Valitutto Giuseppe

Cos' era FULLDENT 1.0

“FullDent 1.0” era un software in grado di gestire le esigenze di un centro odontoiatrico quali: gestione degli appuntamenti, dottori, sale, fornitori. Le prenotazioni potevano essere effettuate esclusivamente sull'unico terminale dove era installato il software quindi, l'utente doveva chiamare o recarsi personalmente al centro per effettuare una prenotazione.

Cosa sarà FULLDENT 2.0

“FullDent 2.0” permetterà all'utente di gestire autonomamente le proprie prenotazioni comodamente dal proprio smartphone. Quest'ultime potranno essere quindi visualizzate dalla segretaria del centro in tempo reale evitando all'utente di recarsi personalmente nello studio oppure di contattare telefonicamente il centro per gestire i propri appuntamenti. Un'agenda pubblica condivisa a cui tutti gli utenti potranno accedere consentirà una gestione più agevole della comunicazione.

Cosa si è scelto di migrare?

-La gestione degli appuntamenti: in “FullDent 1.0” possibile solo dalla segretaria, in “FullDent 2.0” anche da parte dell’utente.

-La gestione dello storico: in “FullDent 1.0” possibile solo dalla segretaria, in “FullDent 2.0” opportunamente ridisegnata per rendere fruibile l’accesso ai dati sia dall’utente che dalla segretaria.

Cosa si è scelto di aggiungere?

-Gestione paziente. Prima della migrazione “FullDent 1.0” non memorizzava alcuna informazione (al di fuori di quelle necessarie per la prenotazione) riguardo l’utente. In “FullDent 2.0” è prevista la gestione dell’anagrafica di ciascun utente in modo da conservare informazioni utili riguardanti ogni singolo paziente.

Perché si è scelto di migrare tali funzionalità?

Il nostro obiettivo era quello di creare un'applicazione che permettesse all'utente di poter effettuare autonomamente una prenotazione quindi le restanti parti del sistema non sono state prese in considerazioni in quanto irrilevanti per il raggiungimento dell'obiettivo prefissatoci.

Quali scelte progettuali sono state vagliate (ed utilizzate)?

E' stato necessario passare da un'architettura centralizzata ad un architettura client-server.

Il codice dell'applicazione da migrare era scritto in Java; per questo motivo, al fine di ridurre i costi di sviluppo, il web-service è stato implementato utilizzando le API JAX-RS di java per la creazione di un servizio REST. Questo ci ha permesso di riutilizzare tutta quella parte di codice delegata alla comunicazione con il database. L'interfaccia grafica dell'applicazione utilizzata dalla segretaria è stata modificata opportunamente in modo tale da renderla più user-friendly.

Il dbms utilizzato resta sempre MySQL. La base di dati utilizzata è stata normalizzata in quanto alcune tabelle non rispettavano i vincoli di integrità referenziale (in particolare 2NF e 3NF).

Le ulteriori tecnologie utilizzate per la realizzazione del progetto sono:

- API Android: necessarie per la realizzazione del client Android utilizzato dall'utente per gestire la comunicazione con il web-service.

- JERSEY: framework Open Source che implementa le specifiche JAX- RS, Java API for RESTful Web Services, utilizzando le annotazioni per mappare una classe Java ad una risorsa Web. All'implementazione di JAX-RS, Jersey aggiunge nuove API per consentirne l'estensione secondo le necessità dello sviluppatore. Supporta nativamente rappresentazioni in JSON, sia come client che come server.

- JSON: formato adatto per lo scambio dei dati in applicazioni client- server. Nella fattispecie JSON è stato scelto principalmente per nascondere i dettagli implementativi del web-service ai client creati. Inoltre, per sviluppi futuri sarà facile integrare client che utilizzano s.o eterogenei (iOS, Windows Phone, BlackBerry) in quanto permetterà di far “dialogare” sistemi scritti in linguaggi di programmazione differenti.

Per il testing del sistema è stato utilizzato:

-Unit Testing: la tecnica scelta si riferisce alla metodologia black box, ovvero, uno sviluppo incentrato sul controllo dell'input e dell'output previsto.

-Integration Testing: è stata usata la metodologia bottom-up, ovvero il testing dei moduli partendo dai livelli più bassi della gerarchia fino all'interfaccia grafica. La lista dei casi di test è stata generata dall'applicazione del metodo Category Partition. L'esecuzione dei casi di test sono stati eseguiti manualmente anziché con il tool Selenium.

Quali altre alternative sono state vagliate e perché sono state scartate?

In un primo momento si era pensato di migrare l'applicazione sul web. Tale alternativa è stata in un secondo momento scartata in quanto l'utilizzo di un app per l'utente è di più facile accesso e di immediato utilizzo rispetto ad una versione mobile di un sito web.

