

Specifica dei Requisiti Neuro Assistant



Settembre 1, 2011
Versione 1.0

Abdul Rasheed Tutakhail

Storico modifiche

Data	Versione	Descrizione	Updated By
Settembre 1,2011	0,1	Versione finale	----

Approvazione documento

Nome	Ruolo	Firma

Contenuti

1	Introduzione	4
1.1	Scopo	4
1.2	Scopo del Progetto.....	4
1.3	Glossario.....	4
1.4	Referenze.....	5
1.5	Panoramica	5
2	Descrizione Generale	6
2.1	System Environment.....	6
2.2	Specifica dei Requisiti	7
2.2.1	Use Case Dottore	7
2.2.2	Use Case Ricercatore	8
3	Panoramica delle Tecnologie utilizzate.....	8
3.1	Web Service	8
3.2	Web Services Standards.....	9
3.2.1	XML.....	9
3.2.2	SOAP	9
3.2.3	WSDL	10
3.2.4	UDDI	10
3.3	Apache Web Server.....	11
3.4	MySQL	11

1 Introduzione

1.1 Scopo

Lo scopo di questo documento è di presentare una descrizione dettagliata del Neuro Assistant. Sarà spiegato lo scopo e le caratteristiche del sistema, le interfacce, ciò che il sistema farà, i vincoli alle quali deve operare e come il sistema reagisce agli stimoli esterni. Questo documento è destinato agli sviluppatori del sistema e sarà proposto come un progetto all'esame del Open Distributed Systems, per la sua approvazione.

1.2 Scopo del progetto

Il sistema software effettuerà il lavoro di un assistente virtuale che aiuterà il medico nel processo di diagnosi di un paziente. Avrà la capacità di proporre una serie di diagnosi secondo le informazioni relative memorizzate nel database e tutto il processo sarà molto efficiente e veloce. Senza l'accesso a Neuro Assistant, i medici devono raccogliere manualmente ed elaborare le informazioni individuali relative a un paziente, e spendere molto tempo per trovare una diagnosi esatta avendo anche molti pazienti da dover trattare soprattutto nei grandi ospedali, mentre Neuro Assistant con la sua interfaccia user-friendly il medico non solo accelera il processo di raccolta dati, ma lo rende anche molto più sicuro.

Implementando il sistema software sarà possibile per i ricercatori con una username e password aggiungere nuove informazioni al database, cosicché il loro lavoro sarà subito fruibile al pubblico.

1.3 Glossario

Termine	Definizione
Medico - Neurochirurgo	Persona che inserisce i parametri per il sistema software e riceve i risultati sotto forma di un elenco di possibili diagnosi
Paziente	Persona i cui dati vengono elaborati dal Sistema Software dopo essere stato introdotto dal Neurochirurgo / medico.
Database	Raccolta di tutte le informazioni del sistema.
Ricercatore	Persona che contribuisce alla gestione, inserimento dei dati nel database
Campo	Una cell in un form.
Review	Una raccomandazione scritta circa l'opportunità di una pubblicazione come articolo, possono includere proposte di miglioramento.
Specificazione dei requisiti software	Un documento che descrive completamente tutte le funzioni del sistema proposto e le costrizioni alle quali deve operare. Per esempio, questo documento.
Stakeholder	Qualsiasi persona con un interesse per il progetto che non è uno sviluppatore.
Utente	Recensore, Medico / Neurochirurgo o Ricercatore.

1.4 Referenze

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

1.5. Panoramica

Il prossimo capitolo, la sezione Descrizione generale, del presente documento offre una panoramica delle funzionalità del prodotto. Esso descrive i requisiti informale e viene utilizzato per stabilire un contesto per le specifiche tecniche richieste nel

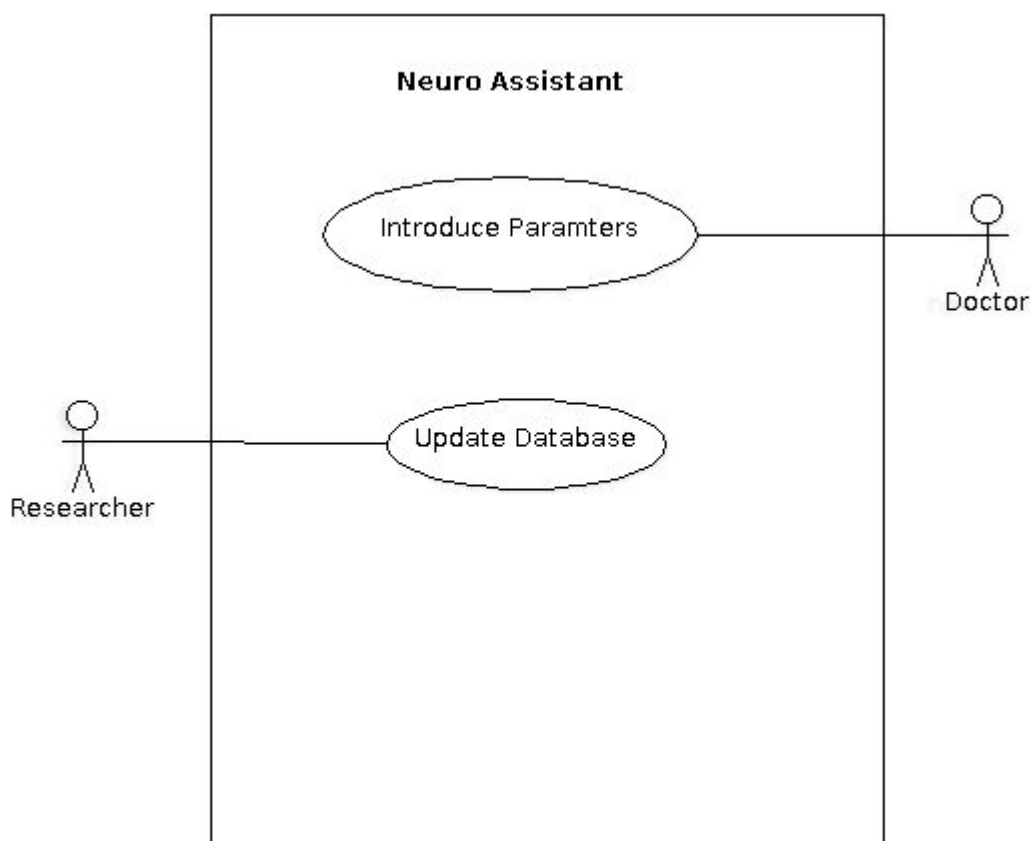
prossimo capitolo.

Il terzo capitolo, sezione specifica dei requisiti, di questo documento è destinato principalmente agli sviluppatori e descrive in termini tecnici, i dettagli delle funzionalità del prodotto.

Entrambe le sezioni del documento descrivono il prodotto software nella sua interezza, ma sono destinati a diversi tipi di pubblico.

2. Descrizione Generale

2.1 System Environment



Neuro Assistant ha due attori attivi, Ricercatore e Dottore. Dopo aver introdotto il loro nome utente e password a seconda dell'attore della scena sarà visualizzata una diversa interfaccia grafica attraverso la quale possono interagire con il sistema.

2.2 Specifica dei requisiti

Questa sezione descrive i casi di utilizzo per ciascuno dei lettori attivi separatamente. Il Dottore e il Ricercatore sono i principali attori in questo sistema.

2.2.1 Use Case Dottore

Use Case: **Introduce parametri**

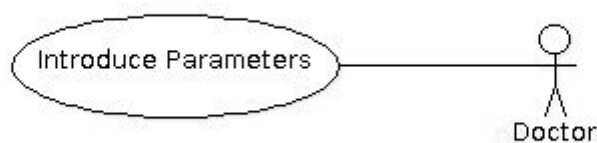


Diagramma:

Breve Descrizione

Il dottore inserisce informazioni su un particolare paziente.

Descrizione passo-passo

Prima di usare questo use case, il dottore ha già avuto accesso alla pagina principale

di Neuro Assistant. Relativo a al Medico.

1. Il dottore inserisce l'età.
2. Il dottore specifica l'etnia del paziente dal menu a tendina che trova sull'interfaccia.
3. Il dottore specifica i sintomi dalla check-list.
4. Il dottore specifica il sesso del paziente dal radio button relativi.
5. Il dottore invia il modulo appena ha compilato tutti i campi.
6. Il sistema processa i dati e stampa il risultato come diagnosi.

2.2.2 Use Case Ricercatore

Use case: **Aggiornamento Database**

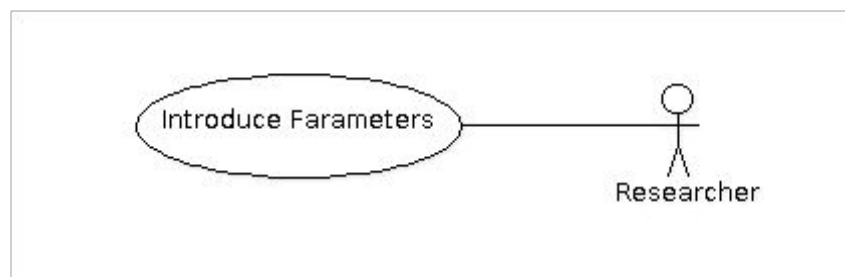


Diagramma:

Breve Descrizione

Il ricercatore mantiene il database di Neuro Assistant aggiornato

Descrizione passo-passo

Prima di questo caso, il ricercatore ha già accesso alla pagina di Neuro Assistant dedicata per aggiungere nuove informazioni nel database.

1. Il ricercatore inserisce il nome della diagnosi.
2. Il ricercatore controlla i sintomi collegati alla diagnosi utilizzando le check-box
3. Il ricercatore può aggiungere ulteriori informazioni relative alla malattia utilizzando il campo descrizione.
4. Il ricercatore specifica l'età delle persone che sono più a rischio dalla data malattia, inserendo un valore nel campo età
5. Il ricercatore specifica le origini delle persone più a rischio della malattia utilizzando le check-box relativi all'etnia.

6. Il ricercatore facendo clic sul pulsante salva aggiunge le informazioni al database.
7. Il ricercatore può inserire un'ulteriore diagnosi cliccando sul pulsante new.

3. Panoramica delle tecnologie utilizzate

3.1 Web Services

Un Web Service (servizio web) è un sistema software progettato per supportare l'interoperabilità tra diversi elaboratori su di una medesima rete; caratteristica fondamentale di un Web Service è quella di offrire un'interfaccia software (descritta in un formato automaticamente elaborabile quale, ad esempio, il Web Services Description Language) utilizzando la quale altri sistemi possono interagire con il Web Service stesso attivando le operazioni descritte nell'interfaccia tramite appositi "messaggi" inclusi in una "busta" (la più famosa è SOAP): tali messaggi sono, solitamente, trasportati tramite il protocollo HTTP e formattati secondo lo standard XML.

Proprio grazie all'utilizzo di standard basati su XML, tramite un'architettura basata sui Web Service (chiamata, con terminologia inglese, Service oriented Architecture - SOA) applicazioni software scritte in diversi linguaggi di programmazione e implementate su diverse piattaforme hardware possono quindi essere utilizzate, tramite le interfacce che queste "espongono" pubblicamente e mediante l'utilizzo delle funzioni che sono in grado di effettuare (i "servizi" che mettono a disposizione) per lo scambio di informazioni e l'effettuazione di operazioni complesse (quali, ad esempio, la realizzazione di processi di business che coinvolgono più aree di una medesima azienda) sia su reti aziendali come anche su Internet: la possibilità dell'interoperabilità fra diversi linguaggi di programmazione (ad esempio, tra Java e Python) e diversi sistemi operativi (come Windows e Linux) è resa possibile dall'uso di standard "aperti".

3.2 Web Services Standards

3.2.1 XML

XML (sigla di **eXtensible Markup Language**) è un metalinguaggio di markup, ovvero un linguaggio marcatore che definisce un meccanismo sintattico che consente di estendere o controllare il significato di altri linguaggi marcatori.

Costituisce il tentativo di produrre una versione semplificata di SGML che consenta di definire in modo semplice nuovi linguaggi di markup da usare in ambito web. Il nome indica quindi che si tratta di un linguaggio marcatore (markup language) estensibile (eXtensible) in quanto permette di creare tag personalizzati.

3.2.2 SOAP

Simple Object Access Protocol è un protocollo leggero per lo scambio di messaggi tra componenti software, tipicamente nella forma di componentistica software. La parola *object* manifesta che l'uso del protocollo dovrebbe effettuarsi secondo il paradigma della programmazione orientata agli oggetti.

SOAP è la struttura operativa (*framework*) estensibile e decentralizzata che può operare sopra varie pile protocollari per reti di computer. I richiami di procedure remote possono essere modellizzati come interazione di parecchi messaggi SOAP. SOAP dunque è uno dei protocolli che abilitano i servizi Web.

SOAP può operare su differenti protocolli di rete, ma HTTP è il più comunemente utilizzato e l'unico ad essere stato standardizzato dal W3C. SOAP si basa sul metalinguaggio XML e la sua struttura segue la configurazione Head-Body, analogamente ad HTML. Il segmento opzionale **Header** contiene meta-informazioni come quelle che riguardano il routing, la sicurezza, le transazioni e parametri per l'Orchestration. Il segmento obbligatorio **Body** trasporta il contenuto informativo e talora viene detto carico utile, o *payload*. Questo deve seguire uno schema definito dal linguaggio XML Schema. Soap può essere utilizzato in due modi diversi per una chiamata: 1) Richiesta via SOAP di parametri: Il client controlla nel Service Registry l'oggetto d'interesse e sviluppa il messaggio secondo i parametri contenuti nel Service Registry. 2) General Purpose Messaging.

Un programmatore può sviluppare un suo protocollo privato, il client conosce a priori i parametri e non necessita di consultare il service registry. All'interno del body del messaggio inserisco i dati scritti nel formato concordato con lo sviluppatore.

3.2.3 WSDL

Il Web Services Description Language (WSDL) è un linguaggio formale in formato XML utilizzato per la creazione di "documenti" per la descrizione di Web Service.

Mediante WSDL può essere, infatti, descritta l'interfaccia pubblica di un Web Service ovvero creata una descrizione, basata su XML, di come interagire con un determinato servizio: un "documento" WSDL contiene infatti, relativamente al Web Service descritto, informazioni su:

- cosa può essere utilizzato (le "operazioni" messe a disposizione dal servizio);

- come utilizzarlo (il protocollo di comunicazione da utilizzare per accedere al servizio, il formato dei messaggi accettati in input e restituiti in output dal servizio ed i dati correlati) ovvero i "vincoli" (bindings in inglese) del servizio;
- dove utilizzare il servizio (cosiddetto endpoint del servizio che solitamente corrisponde all'indirizzo - in formato URI - che rende disponibile il Web Service)

Le operazioni supportate dal Web Service ed i messaggi che è possibile scambiare con lo stesso sono descritti in maniera astratta e quindi collegati ad uno specifico protocollo di rete e ad uno specifico formato.

Il WSDL è solitamente utilizzato in combinazione con SOAP e XML Schema per rendere disponibili Web Services su reti aziendali o su internet: un programma client può, infatti, "leggere" il documento WSDL relativo ad un Web Service per determinare quali siano le funzioni messe a disposizione sul server e quindi utilizzare il protocollo SOAP per utilizzare una o più delle funzioni elencate dal WSDL.

3.2.4 UDDI

L'UDDI (acronimo di Universal Description Discovery and Integration) è un registry (ovvero una base dati ordinata ed indicizzata), basato su XML ed indipendente dalla piattaforma hardware, che permette alle aziende la pubblicazione dei propri dati e dei servizi offerti su internet.

L'UDDI è uno degli standard alla base del funzionamento dei Web Service: è stato progettato per essere interrogato da messaggi in SOAP e per fornire il collegamento ai documenti WSDL che descrivono i vincoli protocollari ed i formati dei messaggi necessari per l'interazione con i Web Service elencati nella propria directory.

3.3 Apache Web Server

Apache HTTP Server, o più comunemente **Apache** è il nome dato alla piattaforma server Web modulare più diffusa (ma anche al gruppo di lavoro *open source* che ha creato, sviluppato e aggiornato il software server), in grado di operare da sistemi operativi UNIX/Linux e Microsoft.

Apache è un software che realizza le funzioni di trasporto delle informazioni, di internetwork e di collegamento, ha il vantaggio di offrire anche funzioni di controllo per la sicurezza come quelli che compie il proxy.

3.4 MySQL

MySQL è un Relational database management system (RDBMS), composto da un client con interfaccia a caratteri e un server, entrambi disponibili sia per sistemi Unix come GNU/Linux che per Windows, anche se prevale un suo utilizzo in ambito Unix.

MySQL come alternativa ai sistemi di database proprietario di Oracle, IBM e Informix. MySQL è attualmente disponibile in due diversi accordi di licenza: gratuito, sotto la GNU General Public License (GPL) del sistema open source. Esistono diversi tipi di MySQL Manager, ovvero di strumenti per l'amministrazione di MySQL. Uno dei programmi più popolari per amministrare i database MySQL è phpMyAdmin (richiede un server web come Apache HTTP Server ed il supporto del linguaggio PHP). Si può utilizzare facilmente tramite un qualsiasi browser.

MySQL è utilizzato in una vasta gamma di applicazioni, tra cui il data warehousing, e-commerce, banche dati Web, applicazioni di registrazione e applicazioni distribuite. E' inoltre sempre più integrate in software di terze parti e altre tecnologie. Secondo MySQL AB, il loro prodotto di punta è di oltre sei milioni di installazioni attive MySQL in tutto il mondo. I clienti includono Cisco, Dun & Bradstreet, Google, NASA, Lufthansa, Hyperion, e Suzuki.

