

MyTalk

Software di comunicazione tra utenti senza
requisiti di installazione



clockworkTeam7@gmail.com

Manuale sviluppatore

v 1.0

Informazioni sul documento

Nome documento	Manuale sviluppatore
Versione documento	v 1.0
Data creazione	2013/02/25
Data ultima modifica	2013/03/26
Uso documento	Esterno
Redazione	Ceseracciu Marco
Verifica	La Bruna Agostino
Approvazione	Gavagnin Jessica
Lista distribuzione	gruppo <i>Clockwork</i> Zucchetti SPA Prof. Tullio Vardanega

Sommario

Il presente documento vuole guidare lo sviluppatore nella espansione del prodotto **MyTalk** esponendo e spiegando le funzionalità offerte.

Diario delle modifiche

Autore	Modifica	Data	Versione
Gavagnin Jessica	Approvazione del documento	2013/03/26	v1.0
La Bruna Agostino	Verifica del documento	2013/03/25	v0.6
Ceseracciu Marco	Stesura capitolo Guida allo sviluppo	2013/03/24	v 0.5
Ceseracciu Marco	Stesura capitolo Configurazione Eclipse	2013/03/24	v 0.3
Ceseracciu Marco	Stesura capitolo Recupero Progetto per sviluppatori	2013/03/34	v 0.2
Ceseracciu Marco	Creazione documento	2013/02/22	v 0.1

Indice

1	Introduzione	1
1.1	Scopo del documento	1
1.2	Descrizione generale	1
1.3	Glossario	1
2	Recupero Progetto per sviluppatori	2
2.1	Abilitazione	2
2.2	Creazione di una copia locale del repository	2
3	Configurazione Eclipse	3
3.1	Preparazione Eclipse	3
3.2	Importazione del progetto in Eclipse	3
4	Guida allo sviluppo	4
4.1	Server	4
4.1.1	Gestione dati	4

Elenco delle figure

Elenco delle tabelle

1 Introduzione

1.1 Scopo del documento

Lo scopo di questo documento è quello di fornire, agli utenti che vogliono partecipare al suo sviluppo, linee guida per l'utilizzo del prodotto **MyTalk**.

1.2 Descrizione generale

Il prodotto **MyTalk** è uno strumento che permette la comunicazione di tipo VoIP (Voice over IP) tra due o più utenti in tempo reale tramite Internet, senza l'installazione di programmi od estensioni¹, così da non comportare costi per l'installazione di strumenti che permettono le stesse funzionalità che le aziende di grandi dimensioni sosterebbero.

1.3 Glossario

Per la spiegazione dei termini tecnici, abbreviazioni ed acronimi presenti all'interno dei documenti, si faccia riferimento al documento esterno **Glossario_v2.0.pdf**, dato in allegato. Come specificato nel documento **Norme_di_Progetto_v3.0.pdf**, verrà sottolineata in ogni documento la prima occorrenza del termine da specificare.

¹Ad eccezione del plugin WebRTC4All.

2 Recupero Progetto per sviluppatori

2.1 Abilitazione

Per accedere al repository per sviluppatori al progetto bisogna richiedere l'abilitazione:

1. Iscrivarsi al sito [GitHub](#)
2. Mandare una mail a clockworkteam7@gmail.com con oggetto **RICHIESTA ABILITAZIONE** riportando nel contenuto il nome utente da abilitare e le motivazioni della richiesta
3. Riceverà una mail di conferma dell'abilitazione dell'account; in caso di rifiuto riceverà una mail con dettagliate le motivazioni del rifiuto

2.2 Creazione di una copia locale del repository

1. Per utilizzare il repository in locale è necessario installare la versione del client git per il proprio sistema operativo². In questa maniera è possibile comunicare e tenere aggiornata la propria copia con quella degli altri sviluppatori del progetto
2. A questo punto è necessario inserire i propri dati di registrazione a Github, inoltre è necessario inserire il percorso in cui posizionare la propria cartella locale del repository
3. Una volta effettuata l'autenticazione si aprirà una schermata, e dopo essere stati abilitati all'utilizzo del repository, si aprirà una finestra che mostra i repository dove si è stati abilitati. A questo punto bisogna evidenziare il repository interessato e cliccare su *clone*. La clonazione del repository avverrà automaticamente nella cartella locale scelta nel punto 2
4. A questo punto il repository locale è sincronizzato con quello online

²GitHub for Windows, Github for Mac o Git-Cola per Linux.

3 Configurazione Eclipse

Il gruppo *Clockwork* consiglia l'utilizzo di IDE Eclipse per lavorare su **MyTalk**. Di seguito viene riportata una breve guida alla configurazione di questo strumento per lavorare al prodotto.

3.1 Preparazione Eclipse

1. Scaricare Eclipse dal sito ufficiale³. Si consiglia la versione per Java Developers⁴
2. Nel caso non sia presente nella versione installata, bisogna scaricare EGit⁵ per gestire il repository
3. Nel caso non sia presente nella versione installata, bisogna scaricare JUnit⁶ per gestire i test

3.2 Importazione del progetto in Eclipse

Per poter lavorare in Eclipse bisogna importare il progetto dal repository:

1. Andare su File -> Import
2. Nella finestra *Select* selezionare Git -> Projects from Git⁷. Premere *Next*
3. Nella finestra *Select Repository Source* selezionare URI. Premere *Next*
4. Nella finestra *Select Repository Source* inserire nel campo URI il link necessario per effettuare lo scaricamento del progetto. I campi necessari una volta inserito l'URI si autocompleteranno, esclusa la password che bisogna inserire manualmente. Premere *Next*
5. Nella finestra *Branch Selection* selezionare *master*. Premere *Next*
6. Nella finestra *Local Destination* premere il bottone *Browse* e selezionare la cartella dove risiede il vostro localhost come cartella di destinazione. Premere *Next*
7. Selezionare poi nella finestra *Select a wizard to use for importing project* l'opzione *Import existing project*. Premere *Next*
8. Controllare che sia selezionato il *Project* corretto e premere *Finish*
9. Il vostro progetto è importato in Eclipse

³<http://www.eclipse.org/downloads/>.

⁴Questa decisione è stata presa poiché la base di sviluppo del prodotto in questione è Java.

⁵<http://www.eclipse.org/egit/>.

⁶<http://github.com/KentBeck/junit/downloads>.

⁷In caso di mancanza di tale opzione assicurarsi che sia installato correttamente EGit.

4 Guida allo sviluppo

Il gruppo *Clockwork*, in fase di progettazione, ha pensato di rendere il prodotto **MyTalk** facilmente estendibile in merito a:

- Database
- Servizi di comunicazione

4.1 Server

Quando il progetto **MyTalk** è iniziato, era richiesto un database leggero, di conseguenza è stato utilizzato SQLite.

Si è deciso dunque di implementare nel package `mytalk.server.dao` la classe `JavaConnectionSQLite`, in maniera da accedere al database solamente da una classe.

Infatti in caso di un futuro cambiamento del database, dovuto a nuove necessità, basterà adattare la classe `JavaConnectionSQLite` per connettersi al nuovo database.

4.1.1 Gestione dati

La gestione dei dati presenti nel database è effettuata dalle classi presenti nel package `mytalk.server.dao`.

In caso di necessità di apportare dei cambiamenti alla gestione dei dati non servirà modificare gli strati superiori, in quanto è stata realizzata una interfaccia per ogni classe, così da evitare di dover riadattare tutti i moduli con cui il dao interagisce.

La natura multiplatforma, e la precisa volontà da parte degli sviluppatori della tecnologia WebRTC di progettare dei protocolli per farli diventare standard rendono **MyTalk** facilmente estensibile, purché si resti all'interno dei limiti delle funzionalità supportate.