

# RUBRICA

## 1 - Introduzione :

L'applicazione è una rubrica privata permette all'utente di registrarsi con nome utente e password , quando si apre l'applicativo l'utente può registrare i contatti per nome, cognome , sesso , data di nascita , numero di telefono , città di residenza , indirizzo di residenza , codice di avviamento postale .

## 2 - Funzionamento del programma :

La prima volta che si lancia l'applicativo viene creato nel sistema una cartella con nome **my rubrica** nel path `/home/user/Documents` in cui si salva un file **Utenti** in formato xml contiene una lista degli utenti che si registrano , e per ogni nuovo utente si registra , l'applicativo crea dentro **my rubrica** una propria cartella ha il nome del utente in cui può salvare i suoi contatti . in questa semplice maniera possiamo separare i file degli utenti .

Per esempio , ci sono tre utenti hanno registrato **mario** , **alessandro** , **francesca** , nella cartella **my rubrica** troviamo il file **Utenti** contiene i nomi utente e le loro password , e tre cartelle con nome **mario** , **alessandro** , **francesca** come mostrato nella figura 1 :

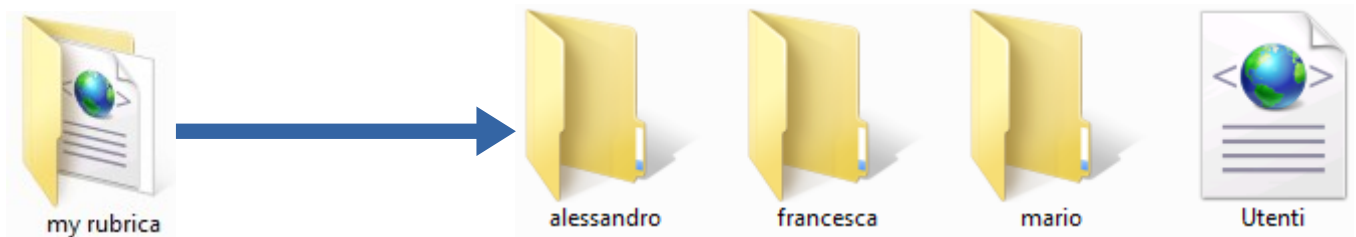


Figura 1

## 3 - Componenti dell'applicazione :

Abbiamo utilizzato il modello **MVC** Model, View, Controller per separare le varie componenti dell'applicazione. Collocata nel pacchetto **rubrica** l'applicazione con il main, **RubricaMain** , e abbiamo:

- \* nel pacchetto **registro.model** le classi che permettono di realizzare i contatti e gli utenti .

- \* nel pacchetto **model.view** i file fxml che permettono di configurare l'interfaccia utente.

\* nel pacchetto `model.controller` le classi con i controlli dell'interfaccia grafica.

Abbiamo due altri pacchetti :

\* `registro.model.util` : contiene le classi che servono a gestire le date, il formato xml e i filtri per l'estrazione dei dati.

\* `images` : contiene tutti le immagine e le iconne che abbiamo utilizzato nell'applicazione .

