

UNIVERSITÀ DEGLI STUDI DI MESSINA

CORSO DI LAUREA TRIENNALE IN INFORMATICA



PROGETTO DI ING. DEL SOFTWARE

Relazione del Progetto: College Social Network

Il progetto verte nello sviluppare una rete sociale all'interno di un college/università in modo che studenti, professori e ricercatori possano scambiarsi idee e conoscenze.

Studente: Pierpaolo Gumina (mat. 454055)

Docente: Prof. S. Distefano

Anno Accademico: 2018/19

Indice.

PREFAZIONE:

- 1. DESCRIZIONE DEL PROBLEMA (RAPPORTO CON IL CLIENTE).....2*
- 2. METODOLOGIA DI SVILUPPO.....3*

CAPITOLO 1: SVILUPPO DEL COLLEGE SOCIAL NETWORK

- 1. LOGIN/HOME PAGE.....5*
 - 1.1. MODELLAZIONE*
 - 1.2. IMPLEMENTAZIONE*
 - 1.3. TEST*
- 2. POST E COMMENTI.....8*
 - 2.1. MODELLAZIONE*
 - 2.2. IMPLEMENTAZIONE*
 - 2.3. TEST*
- 3. MESSAGGI PRIVATI.....9*
 - 3.1. MODELLAZIONE*
 - 3.2. IMPLEMENTAZIONE*
 - 3.3. TEST*
- 4. PROFILO/MODIFICA PROFILO.....11*
 - 4.1. MODELLAZIONE*
 - 4.2. IMPLEMENTAZIONE*
 - 4.3. TEST*

CAPITOLO 2: INTRODUZIONE DELLE FIGURE PROFESSORI E DEL CONCETTO DI CANALI

- 1. LOGIN PROFESSORI/HOME PAGE STUDENTI RIVISITATA.....13*
 - 1.1. MODELLAZIONE*
 - 1.2. IMPLEMENTAZIONE*
 - 1.3. TEST*
- 2. CANALI E GESTIONE DEGLI STESSI.....15*
 - 2.1. MODELLAZIONE*
 - 2.2. IMPLEMENTAZIONE*
 - 2.3. TEST*

3.	CANALI: POST E COMMENTI.....	17
3.1.	MODELLAZIONE	
3.2.	IMPLEMENTAZIONE	
3.3.	TEST	

CAPITOLO 3: INTRODUZIONE DELLE FIGURE ADMIN

1.	LOGIN ADMIN/ HOME PAGE ADMIN.....	19
1.1.	MODELLAZIONE	
1.2.	IMPLEMENTAZIONE	
1.3.	TEST	
2.	CREAZIONE PROFESSORI/ GESTIONE DEGLI INSCRITTI.....	22
2.1.	MODELLAZIONE	
2.2.	IMPLEMENTAZIONE	
2.3.	TEST	

CAPITOLO 4: RELAZIONI

1.	RICHIESTE DI AMICIZIA.....	24
1.1.	MODELLAZIONE	
1.2.	IMPLEMENTAZIONE	
1.3.	TEST	
2.	SUGGERIMENTO DI AMICIZIE.....	26
2.1.	MODELLAZIONE	
2.2.	IMPLEMENTAZIONE	
2.3.	TEST	

CAPITOLO 5: COMPRAVENDITA LIBRI, BOOKCROSSING E CONDIVISIONE DI APPUNTI

1.	COMPRAVENDITA LIBRI.....	29
1.1.	MODELLAZIONE	
1.2.	IMPLEMENTAZIONE	
1.3.	TEST	
2.	BOOKCROSSING.....	33
2.1.	MODELLAZIONE	
2.2.	IMPLEMENTAZIONE	
2.3.	TEST	
3.	CONDIVISIONE APPUNTI.....	38
3.1.	MODELLAZIONE	
3.2.	IMPLEMENTAZIONE	
3.3.	TEST	

PREFAZIONE.

1. DESCRIZIONE DEL PROBLEMA (RAPPORTO CON IL CLIENTE):

L'evoluzione di internet e i diversi studi hanno portato allo sviluppo di varie piattaforme sociali. Dette appunto "Social Network".

"Ma cos'è realmente un Social Network?"

Il significato è "Rete Sociale". Per rete sociale si intende un gruppo di persone connesse tra di loro sia a livello umano che a livello virtuale. Essa può quindi essere intesa come una rete di persone legate tra loro tramite vincoli familiari, lavorativi, amichevoli e anche come una rete di persone connesse tra loro soltanto virtualmente e senza alcun legame fisico e di conoscenza.

"Ma come funziona un social network?"

Lo scopo principale è lo scambio di conoscenza e informazioni mediante strumenti appositi sviluppati in modo che siano di facile utilizzo ed intuitivi. Non importa che sia uno studente, un professore, un ricercatore o un dipendente tutti hanno la possibilità di divulgare il loro pensiero mettendosi al pari livello di tutti.

"Come si accede ad un social network?"

Per accedere ai servizi offerti da un social network non bisogna disporre di particolari condizioni. Se si sta accedendo per la prima volta occorrerà registrarsi, e l'unico campo richiesto è quello di possedere una email (indirizzo Posta Elettronica).

La fase di registrazione consiste nella compilazione di un proprio profilo, necessario per potersi identificare all'interno del Social Network. Tale profilo comprende: Nome, Cognome, Password, Email, Paese, Sesso, Data di Nascita.

Una volta superata la fase di registrazione è possibile fare il login; basterà inserire email e password, in modo tale che l'autenticazione possa essere univoca e riservata al solo utente registrato.

Effettuato l'accesso l'utente sarà accolto nella pagina principale detta per l'appunto, Home Page, la quale conterrà gli ultimi Post pubblicati dai membri della piattaforma. L'utente così potrà decidere di leggere ogni post ed eventualmente esprimere il proprio pensiero a riguardo mediante un commento, il quale sarà visibile a tutti.

Nella medesima schermata l'utente può anche scrivere un proprio post e, qualora lo volesse, allegare un'immagine; Premendo su invia sarà pubblicato e reso fruibile da tutti i membri del Social. Dopo la pubblicazione, semmai l'utente volesse modificare/cancellare un proprio Post, potrà farlo attraverso la schermata "Profilo".

Ogni membro disporrà di un proprio profilo che potrà personalizzare inserendo una serie di dati tra cui: un'immagine di profilo, un'immagine di copertina, un user-name, una descrizione testuale su di sé, e altro (descritto nello specifico nel paragrafo: *l'analisi dei requisiti*).

La piattaforma dispone di un sistema di messaggistica che permette di mantenere in contatto i membri; I messaggi sono visti esclusivamente dal mittente e dal ricevente così da mantenere la Privacy e consentire la divulgazione esclusiva e riservata ad un solo Utente.

Per lo sviluppo di questo progetto verranno utilizzati i seguenti strumenti:

- ❖ JetBrains PhpStorm (v. 2018.3.5) per la scrittura del codice CSS, HTML, JavaScript, PHP;
- ❖ MAMP (Mac OS, Apache, MySQL, PHP) per il Web Server;
- ❖ per disegnare i grafici UML (Unified Modeling Language);
- ❖ Browser Google Chrome per accedere all'indirizzo della pagina in Local Host.

1.2. *METODOLOGIA DI SVILUPPO:*

Per il progetto ho optato per l'utilizzo dell'Agile Unified Process (AUP). Essa descrive un approccio allo sviluppo di applicazioni software, semplice, facile da comprendere e che utilizza tecniche e concetti agili pur rimanendo fedele al processo RUP. Scott Ambler ha cercato di mantenere Agile UP il più semplice possibile, sia nell'approccio che nella sua descrizione.

L'AUP applica le tecniche di sviluppo agile tra (TDD), Agile Modeling, gestione agile del cambiamento e refactoring del database per migliorare la produttività.

Essa si basa su l'utilizzo di sette pratiche:

1. Modellazione. Comprendere il business dell'organizzazione, il dominio del problema affrontato dal progetto, e individuare una soluzione praticabile per affrontare il dominio del problema.
2. Implementazione. Trasformare i modelli in codice eseguibile ed effettuare un livello base di test, in particolare test unitari.
3. Test. Eseguire una valutazione oggettiva per garantire la qualità. Questo include trovare difetti, la convalida che il sistema abbia funzionato come previsto, e verificare che i requisiti siano stati soddisfatti.
4. Deployment. Pianificare la consegna del sistema ed eseguire tale piano per rendere il sistema disponibile agli utenti finali.
5. Configuration Management. Gestire gli artefatti di progetto. Questo include non solo il tener traccia delle versioni degli artefatti nel tempo ma anche controllarne e gestirne le modifiche.
6. Project Management. Dirigere le attività che si svolgono nell'ambito del progetto. Questo comprende la gestione dei rischi, dirigere le persone (assegnazione di compiti, monitorare i progressi, ...), e il coordinamento con persone e sistemi esterni alla portata del progetto stesso, per essere sicuri che sia consegnato nei tempi e nel budget previsti.
7. Ambienti. Sostenere il resto del progetto, garantendo che un processo corretto, orientamenti (standard e linee guida) e strumenti (hardware, software, ecc) siano disponibili per il team in base alle esigenze.

CAPITOLO 1: SVILUPPO DEL COLLEGE SOCIAL NETWORK

1.1. LOGIN/HOME PAGE: MODELLAZIONE

Prima di inoltrarci attraverso il capitolo 1 è opportuno dare uno sguardo d'insieme ad i casi d'uso (vedi Figura 1).

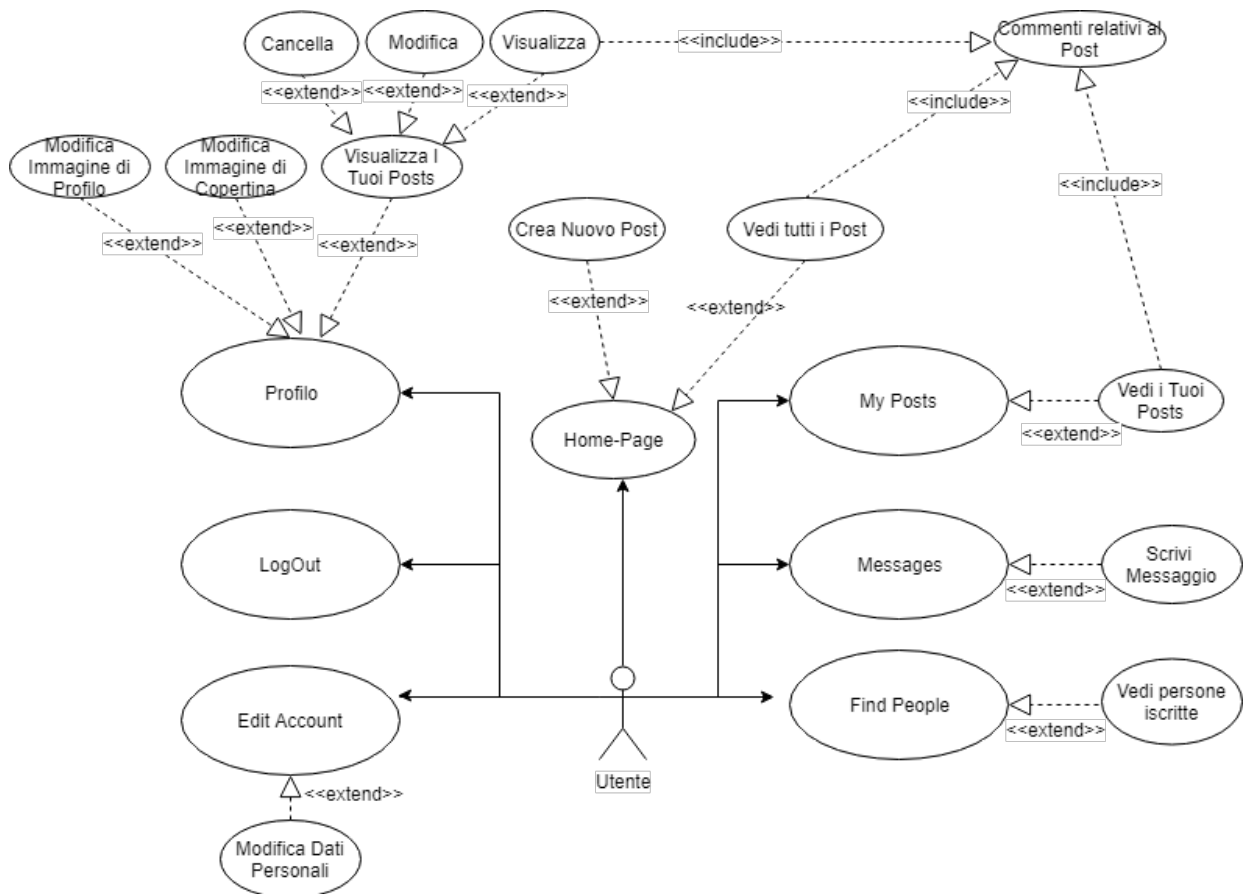


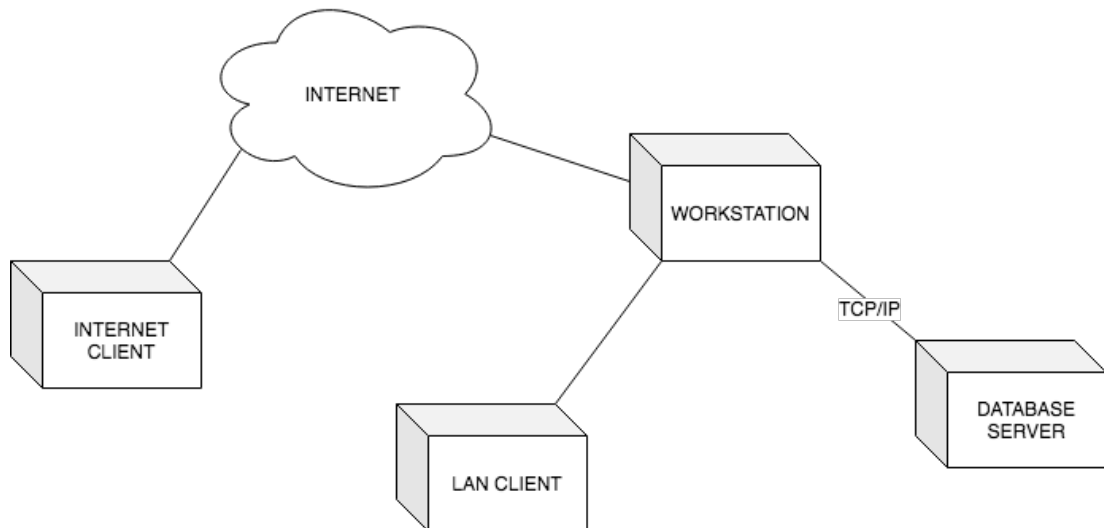
Figura 1. Diagramma dei casi d'uso

Andiamo ad analizzare la fase di Login, indispensabile per una corretta progettazione. Quando l'utente accederà al sito si troverà la schermata di login; In questa fase si hanno due possibili scenari:

- ❖ L'utente, che accede per la prima volta, dovrà effettuare la registrazione;
- ❖ L'utente, che ha già effettuato precedentemente la registrazione, dovrà semplicemente accedere con le proprie credenziali.

La fase di registrazione prevede che un utente (che accede per la prima volta) inserisca alcune generalità tra cui: nome, cognome, sesso, giorno e paese di nascita ed altri campi tra cui: email, password e corso di laurea ove è iscritto.

Al termine dell’inserimento di tutti i campi, premendo il tasto “Iscriviti” l’utente verrà aggiunto nella piattaforma, per accedere quindi basterà recarsi nella pagina “accedi”, nella quale si dovrà inserire email e password.



Deployment Diagram

1.2. LOGIN/HOME PAGE: IMPLEMENTAZIONE

Mostriamo quanto detto nella fase di Modellazione tramite un Activity Diagram (vedi Figura 2).

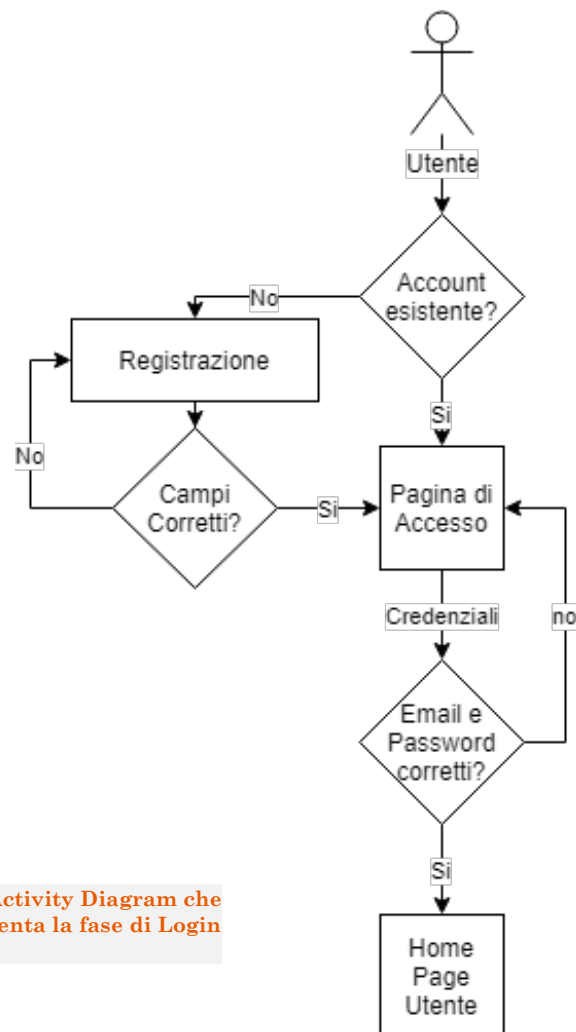


Figura 2. Activity Diagram che rappresenta la fase di Login

Come è possibile notare dalla Figura 2, nel percorso per arrivare alla Home Page sono disseminati vari controlli (rappresentati mediante i rombi), nella quale si esaminano attentamente i dati inseriti dall'utente e solo se quest'ultimi sono corretti è possibile procedere.

Questi controlli non sono altro che risultati di query, fatte tramite PHP e SQL al Database chiamato "Social Network".

Nelle immagini qui sotto è possibile notare la schermata di Registrazione (a destra) e la schermata di Login (a sinistra).

Figura 3. Schermata di Registrazione

Figura 4. Schermata di Login

1.3. LOGIN/HOME PAGE: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Login/Accesso con credenziali errate;	✗
Login/Accesso con credenziali corrette;	✓
Registrazione account senza inserimento di dati obbligatori/E-mail già esistente;	✗
Registrazione account con inserimento di dati obbligatori;	✓
Login/Accesso con Server nello stato di: Down;	✗
Login/Accesso con Server nello stato di: Connesso.	✓

2.1. POST E COMMENTI: MODELLAZIONE

Una caratteristica molto importante nei social network è la possibilità di scrivere e pubblicare propri commenti, stati d'animo, avvenimenti o semplicemente divulgare informazioni. Una volta pubblicato il post potrà essere letto da tutti i membri della piattaforma e se necessario gli utenti potranno inserire un proprio commento, visibile immediatamente sotto il post.

La struttura di un Post può essere di 3 tipologie:

- ❖ Solo testo;
- ❖ Solo immagine;
- ❖ Testo e immagine.

Una volta creato il Post sarà mostrato nella Home-Page e potrà essere letto da tutti i membri della piattaforma, inoltre l'utente che lo desidera ha la possibilità di inserire un proprio commento.

Figura 5. Schermata che illustra il metodo di creazione di un Post

2.2. POST E COMMENTI: IMPLEMENTAZIONE

L'implementazione consiste in un box, costituito da un <div>, all'interno del quale l'utente ha la possibilità di scrivere; Ovviamente l'utente ha un limite di caratteri inseribili, questo limite è di 200 caratteri.

Qualora l'utente superasse tale limite verrà mostrato un Pop-up di errore, contenente il suggerimento di rimuovere dei caratteri.

Il tasto arancione a destra del box (vedi Figura 5), permette all'utente di caricare un'immagine.

Premendo il tasto "Condividi" i dati inseriti nel box verranno caricati nel Database e questa operazione sarà seguita da un Pop-up che darà esito positivo se i dati inseriti sono corretti altrimenti darà esito negativo poiché i dati immessi erano scorretti o si era verificato un altro genere di problema.

Risulta di significativa importanza il campo “id_post” associato ad ogni commento grazie al quale è possibile identificare a quale post esso appartenga.

Il DB (Database) viene così aggiornato con le modifiche appena descritte e implementate, (vedi Figura 6).

Figura 6. Schermata che illustra lo sviluppo del Database fino ad ora

social_network prova	
id_utente	int(30)
nome	varchar(30)
cognome	varchar(30)
user_name	varchar(30)
pass	varchar(30)
email	varchar(30)
paese	varchar(30)
Sesso	varchar(30)
compleanno	date
u_status	varchar(30)
u_posts	varchar(30)
descrizione	varchar(100)
relazioni	varchar(30)
user_cover	varchar(30)
user_image	varchar(30)
user_reg_date	date
recovery_account	varchar(30)
facolta	varchar(20)
grado	varchar(20)

social_network posts	
post_id	int(11)
user_id	int(11)
post_content	varchar(200)
upload_img	varchar(255)
post_date	date

social_network comments	
com_id	int(11)
post_id	int(11)
user_id	int(11)
comment	varchar(255)
comment_author	varchar(255)
date	date

2.3. POST E COMMENTI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Inserimento testo maggiore di 200 caratteri;	✗
Inserimento testo minore di 200 caratteri;	✓
Inserimento di un'immagine non valida;	✗
Inserimento di un'immagine valida;	✓
Premere il tasto “Condividi” senza aver inserito né testo né immagine	✗
Premere il tasto “Condividi” (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Premere il tasto “Condividi” con Server nello stato di: Down.	✗

3.1. MESSAGGI PRIVATI: MODELLAZIONE

Una caratteristica molto importante all'interno di un Social Network è dare la possibilità agli utenti iscritti di contattarsi privatamente tramite messaggi. Bisogna quindi disporre di un meccanismo capace di recapitare messaggi creati da una figura chiamata Mittente e ricevuti da una figura chiamata Destinatario. Innanzitutto è necessario che entrambe le figure Mittente e Destinatario siano correttamente iscritte e rintracciabili all'interno della piattaforma. Successivamente il Mittente recandosi nell'apposito menù di ricerca verificherà tra gli utenti iscritti quale possa essere il Destinatario corrispondente per selezionarlo e scrivergli il messaggio.

Il limite massimo di caratteri è impostato a 200, qualora l'utente ne scrivesse di più verrà mostrato un pop-up di errore d'inserimento.

3.2. MESSAGGI PRIVATI: IMPLEMENTAZIONE

Il sistema si basa sullo scambio di messaggi in modo asincrono, dopo l'invio di un messaggio è necessario fare un refresh di pagina.

Questa scelta agevola notevolmente la realizzazione contrariamente ai sistemi di messaggistica istantanea dove i messaggi vengono inviati ad un server fittizio, ove le due parti (Mittente e Destinatario) interrogano ad intervalli regolari (in genere molto bassi) tale server per verificare la disponibilità di nuovi messaggi.

Disponendo quindi solo di un server in locale, si è preferito utilizzare una tabella dove ogni riga corrisponde ad un messaggio.

Ogni messaggio deve tenere traccia di alcune informazioni base: l'id dell'utente che lo crea, l'id dell'utente a cui il messaggio è destinato, il corpo e la data di creazione.

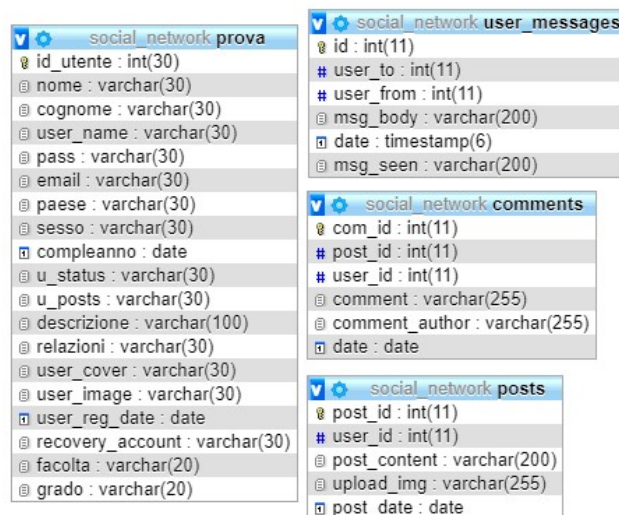


Figura 7. Schermata che illustra lo sviluppo del Database fino ad ora

3.3. MESSAGGI PRIVATI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Inserimento testo maggiore di 200 caratteri;	✗
Inserimento testo minore di 200 caratteri;	✓
Premere il tasto "Invia" senza aver inserito testo;	✗
Premere il tasto "Invia" (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Premere il tasto "Invia" con Server nello stato di: Down.	✗

4.1. PROFILO/MODIFICA PROFILO: MODELLAZIONE

La possibilità di modificare ed aggiornare il proprio profilo utente è di notevole importanza specie se la permanenza all'interno del Social Network potrebbe protrarsi per molti anni.

È quindi importante dare la possibilità all'utente di modificare i propri dati a piacimento secondo il proprio stato d'animo.

Recandosi nell'apposito menù "Edit Account" ogni iscritto avrà alla possibilità di modificare: Nome, Cognome, Username, Descrizione, Password, email, Paese, Sesso e Data di nascita (vedi Figura 8).

Modifica il tuo Profilo	
Cambia il tuo Nome	<input type="text" value="Pierpaolo"/>
Cambia il tuo Cognome	<input type="text" value="Gumina"/>
Cambia il tuo User-Name	<input type="text" value="pierpaolo_gumina_200190"/>
Descrizione	<input type="text" value="Descrizione di default, iscriviti la tua descrizione!"/>
Stato Relazioni	<input type="text" value="..."/>
Password	<input type="password"/> <input type="checkbox"/> Mostra Password
Email	<input type="text" value="pier@gmail.com"/>
Paese	<input type="text" value="Italia"/>
Sesso	<input type="text" value="Maschio"/>
Compleanno	<input type="text" value="10/06/2019"/>
Password Dimenticata	<input type="button" value="Attiva"/>
<input type="button" value="Update"/>	

Figura 8. Schermata che illustra l'operazione di Edit Account

4.1. PROFILO/MODIFICA PROFILO: IMPLEMENTAZIONE

Per applicare quanto detto nella fase di modellazione è sufficiente fare accesso al Database in due fasi:

- ❖ La fase di estrapolazione;
- ❖ La fase di aggiornamento.

La prima fase (estrapolazione) consiste nell'interrogare il DB per ottenere le informazioni che andranno inserite nei rispettivi campi.

La seconda fase (aggiornamento) dopo aver apportato le modifiche al profilo, premendo il tasto "Update" i dati inseriti aggiorneranno la tabella Utenti.

Ovviamente sono presenti dei controlli affinché l'utente non inserisca campi errati, segnalati con dei pop-up di errore.

4.1. PROFILO/MODIFICA PROFILO: TEST

Tipo di Test Effettuato	Esito
Inserimento nome non valido;	✗
Inserimento nome valido;	✓
Lasciare campi vuoti;	✗
Premere il tasto "Update" (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Premere il tasto "Update" con Server nello stato di: Down.	✗

CAPITOLO 2: INTRODUZIONE DELLE FIGURE PROFESSORI E DEL CONCETTO DI CANALI

Dopo una fase di dialogo il cliente ha proposto di implementare delle particolari Features che potessero in qualche modo diversificare tale progetto da un qualsiasi Social Network; Ecco che ho provveduto ad inserire le figure Professori, il contetto dei Canali e delle figure amministratoriali chiamate per l'appunto Admin, con lo scopo di gestire gli utenti all'interno del social. Il seguente capitolo andrà ad illustrare le figure Professori e il contetto di Canali.

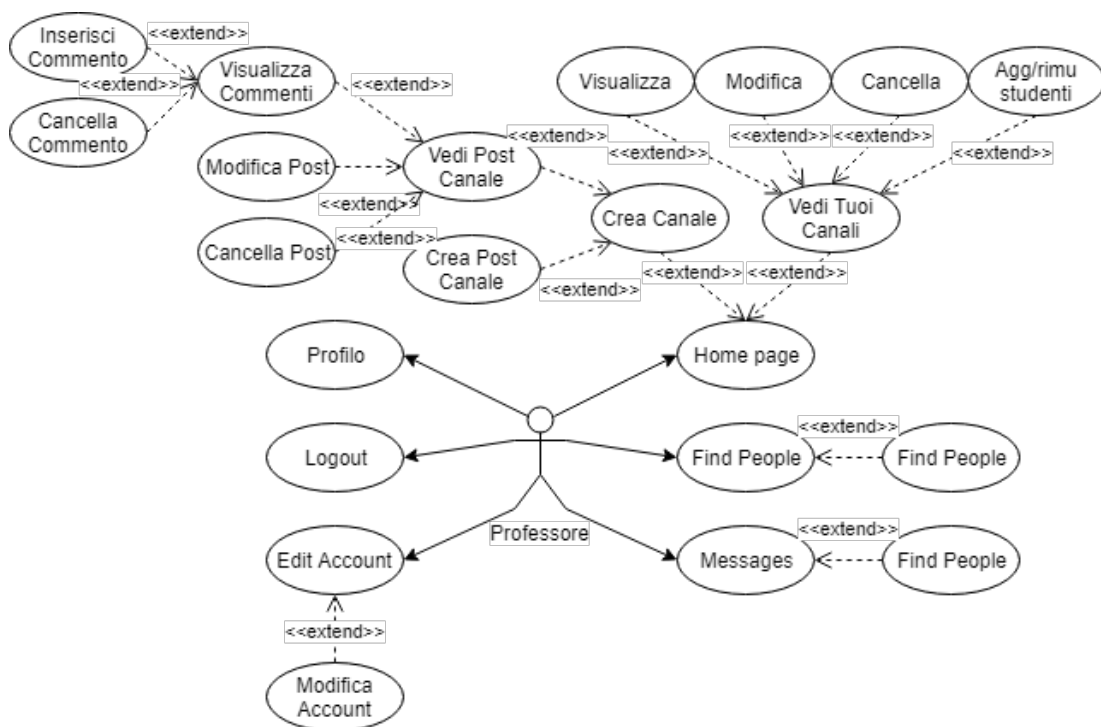


Figura 9. Diagramma dei casi d'uso dal punto di vista dei professori

1.1. LOGIN PROFESSORI/HOME PAGE STUDENTI RIVISITATA: MODELLAZIONE

Prima di inoltrarci nel capitolo 2 è opportuno dare uno sguardo di insieme ad i casi d'uso che ogni professore possiede (vedi Figura 9).

Per le differenti proprietà intrinseche che studenti e professori possiedono si è preferito differenziare la schermata di Login.

Nella precedente schermata studenti è stato aggiunto il tasto "Acesso Admin e Professori" che reindirizza alla schermata dei professori (vedi Figura 10).

N.B. E' necessario che il professore possieda già le sue credenziali, quest'ultime possono essere create solo dall'amministratore per essere fornire al professore (approfondiremo ciò nel capitolo 3).

Sono state apportate importanti modifiche anche alla Home Page degli studenti che adesso fornisce (in alto) le notizie caricate direttamente dal portale della propria facoltà.



Figura 11. Activity Diagram che rappresenta la fase di Login dei Professori

1.2. LOGIN PROFESSORI/HOME PAGE STUDENTI RIVISITATA: IMPLEMENTAZIONE

Mostriamo quanto detto nella fase di Modellazione tramite un Activity Diagram (vedi Figura 11).

Figura 10. Figura che mostra la schermata di login per Professori e Amministratori

1.3. LOGIN PROFESSORI/HOME PAGE STUDENTI RIVISITATA: TEST

Conclusasi la fase di Implementazione sono state eseguite varie fasi di Testing per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Inserimento credenziali errate;	✗
Inserimento credenziali corrette;	✓
Premere il tasto "Accedi" senza aver inserito le credenziali;	✗
Premere il tasto "Accedi" (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Premere il tasto "Accedi" (rispettando i requisiti) con Server nello stato di: Down.	✗

2.1. CANALI E GESTIONE DEGLI STESSI: MODELLAZIONE

Una volta implementate le figure professori è necessario che quest'ultimi possano gestire canali all'interno della piattaforma.

Ispiratomi ai Canali presenti in Telegram, ho ideato un meccanismo dove i professori hanno la possibilità di creare e gestire dei propri canali.

Ogni canale una volta creato andrà popolato con degli studenti, quest'ultimi inseriti esclusivamente dal professore (ovvero gli studenti non potranno inserirsi da soli).

All'interno del canale il professore potrà inserire dei post e potrà gestirli.

Gli studenti inseriti potranno consultare tali post ed eventualmente commentarli.

Il professore, che è il creatore del canale, assume anche le vesti di moderatore: potrà eliminare eventuali commenti inopportuni inseriti all'interno dei post dai vari studenti.

Gli studenti iscritti potranno verificare i contenuti dal menù in alto "I Tuoi Canali".

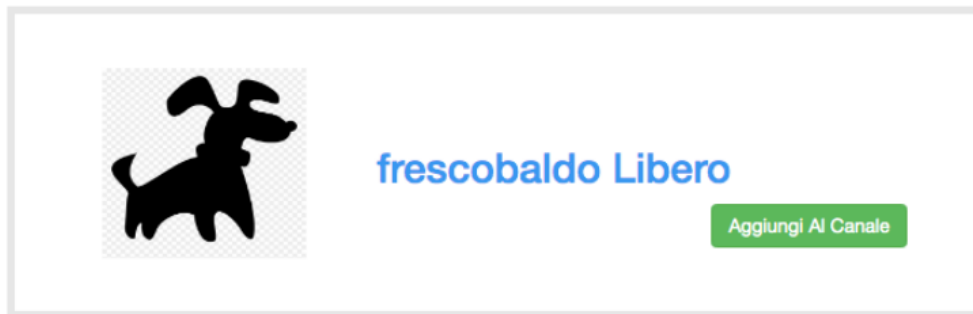
2.2. CANALI E GESTIONE DEGLI STESSI: IMPLEMENTAZIONE

Il professore ha pieno potere sulla gestione d'ogni aspetto del Canale: Creazione, Modifica, Cancellazione, Inserimento e Rimozione di studenti (vedi Figura 11 e 12).



Figura 11. Figura che mostra la schermata con i canali creati e la relativa gestione

Studenti Non Inseriti



Studenti Inseriti

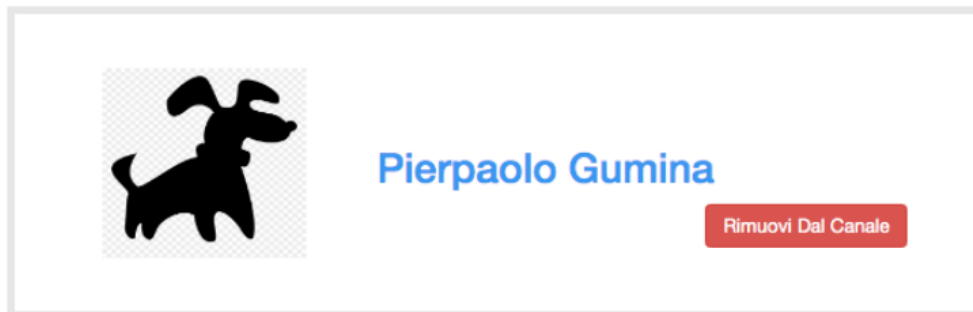


Figura 12. Figura che mostra la gestione degli iscritti al canale

2.3. CANALI E GESTIONE DEGLI STESSI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Creazione Canale con campi errati;	✗
Creazione Canale con campi corretti;	✓
Premere il tasto “Crea” senza aver inserito nulla;	✗
Premere il tasto “Crea” inserendo i campi correttamente;	✓
Provare a creare Canali (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Provare a creare Canali (rispettando i requisiti) con Server nello stato di: Down.	✗

3.1. CANALI: POST E COMMENTI: MODELLAZIONE

I canali costituiscono la forma più intuitiva per gestire gruppi di studenti e veicolare le informazioni dal Professore rivolte allo Studente.

Queste informazioni sono strutturate in dei post pubblicati dal professore, con la possibilità che ogni studente possa inserire un proprio commento.

Il professore, mediante gli strumenti resi disponibili, ha la possibilità di moderare eventuali commenti inopportuni o fuorvianti.

3.2. CANALI: POST E COMMENTI: IMPLEMENTAZIONE



Figura 13. Figura che mostra i post inseriti dal professore

social_network prova id_utente : int(30) nome : varchar(30) cognome : varchar(30) user_name : varchar(30) pass : varchar(30) email : varchar(30) paese : varchar(30) sesso : varchar(30) compleanno : date u_status : varchar(30) u_posts : varchar(30) descrizione : varchar(100) relazioni : varchar(30) user_cover : varchar(30) user_image : varchar(30) user_reg_date : date recovery_account : varchar(30) facolta : varchar(20) grado : varchar(20)	social_network post_canali id_post : int(11) id_canale : int(11) titolo : varchar(50) contenuto : varchar(255) post_img : varchar(255) date : date	social_network commenti_canali id_commento : int(11) post_id : int(11) user_id : int(11) contenuto : varchar(255) data : date	social_network canali id_canale : int(11) nome : varchar(30) id_prof : int(11) data : date titolo : varchar(255)
	social_network professori id_prof : int(11) nome : varchar(20) cognome : varchar(20) password : varchar(20) corso : varchar(30) data : date email : varchar(30)	social_network user_messages id : int(11) user_to : int(11) user_from : int(11) msg_body : varchar(200) date : timestamp(6) msg_seen : varchar(200)	social_network posts post_id : int(11) user_id : int(11) post_content : varchar(200) upload_img : varchar(255) post_date : date
	social_network comments com_id : int(11) post_id : int(11) user_id : int(11) comment : varchar(255) comment_author : varchar(255) date : date	social_network mebri_canali id_m : int(11) id_studente : int(11) id_canale : int(11)	

Il DB con le modifiche apportate in questo capitolo, risulta come in figura 14.

Figura 14. Figura che mostra il Database finora

3.3. CANALI: POST E COMMENTI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Creazione Post Canale con campi errati;	✗
Creazione Post Canale con campi corretti;	✓
Premere il tasto “Crea Post” senza aver inserito nulla;	✗
Premere il tasto “Crea Post” inserendo i campi correttamente;	✓
Provare a creare Post (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Provare a creare Post (rispettando i requisiti) con Server nello stato di: Down.	✗

CAPITOLO 3: INTRODUZIONE DELLE FIGURE ADMIN

Prima di inoltrarci attraverso il capitolo 3 è opportuno dare uno sguardo di insieme ad i casi d'uso cui ogni amministratore possiede (vedi Figura 15).

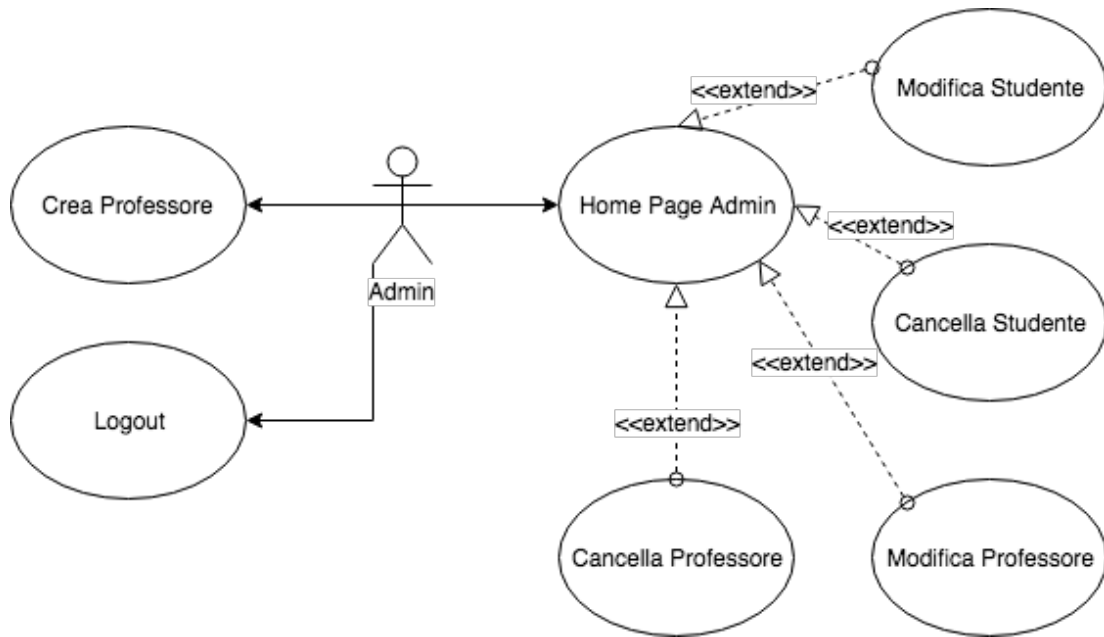


Figura 15. Figura che mostra i casi d'uso degli amministratori

1.1. LOGIN ADMIN / HOME PAGE ADMIN: MODELLAZIONE

Per una corretta gestione della piattaforma è necessario stabilire delle figure: gli amministratori che hanno come unico scopo quello della gestione (modifica, cancellazione, creazione) degli iscritti.

Per la pagina di Login ho provveduto a sfruttare la pagina di accesso dei professori con la condizione di spuntare una casella nel caso in cui l'accesso si stia effettuando come amministratore (vedi Figura 10).

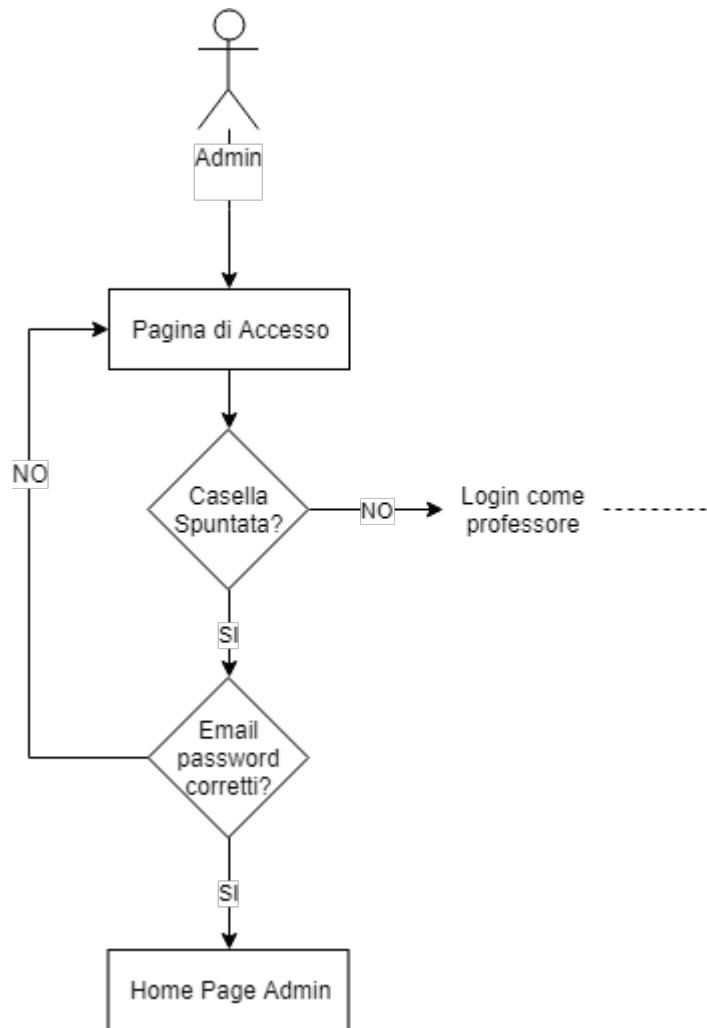


Figura 15. Activity Diagram che rappresenta la fase di Login Amministratori

1.2. LOGIN ADMIN / HOME PAGE ADMIN: IMPLEMENTAZIONE

La Home page è strutturata in due categorie, dove prima vengono mostrati tutti gli studenti e successivamente tutti i professori. Inoltre all’inizio della categoria professori è presente il tasto “Registra Professore” con il quale è possibile inserire una nuova figura professore.

Studenti

**Pierpaolo
Gumina**
Ingegneria - pier@gmail.com - Italia -
Maschio
Registrato il: 2019-06-11

Modifica Account

Cancella Account

**frescobaldo
Libero**
Ingegneria - frescolibero@gmail.com
- Italia - Maschio
Registrato il: 2019-06-11

Modifica Account

Cancella Account

Professori

Registra Professore

Alberto Albertini
Ingegneria -
alberto@gmail.com
Registrato il: 2019-06-11

Modifica Account

Cancella Account

1.3. LOGIN ADMIN / HOME PAGE ADMIN: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Login con credenziali errate;	✗
Login con credenziali corrette;	✓
Login con credenziali corrette ma senza la casella spuntata;	✗
Login con credenziali corrette e casella spuntata;	✓
Login (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Login (rispettando i requisiti) con Server nello stato di: Down.	✗

2.1. CREAZIONE PROFESSORI / GESTIONE DEGLI INSCRITTI: MODELLAZIONE

Oltre alla gestione degli account gli amministratori hanno il compito di creare e quindi fornire le credenziali al professore.

Ho scelto di delegare questo compito agli amministratori poiché i professori hanno un ruolo importate all'interno della piattaforma. Nessuno deve avere la possibilità, seppur remota, di fingersi un professore altrimenti si potrebbe incorrere in gravi problemi di sicurezza informatica.

Ecco che quindi l'Admin provvederà a creare il professore e successivamente fornirgli le credenziali di accesso (nel modo più sicuro possibile).

2.2. CREAZIONE PROFESSORI / GESTIONE DEGLI INSCRITTI: IMPLEMENTAZIONE

Registra Professore

Nome

Cognome

Password

Ingegneria

Email

Registra

Figura 17. Figura che mostra la pagina Registra Professori degli Amministratori

Diamo quindi uno sguardo al Database completo fino ad ora.

social_network canali id_canale : int(11) nome : varchar(30) id_prof : int(11) data : date titolo : varchar(255)	social_network user_messages id : int(11) user_to : int(11) user_from : int(11) msg_body : varchar(200) date : timestamp(6) msg_seen : varchar(200)	social_network amministratori id_admin : int(11) nome : varchar(30) cognome : varchar(30) password : varchar(30) data : date email : varchar(50)
social_network prova id_utente : int(30) nome : varchar(30) cognome : varchar(30) user_name : varchar(30) pass : varchar(30) email : varchar(30) paese : varchar(30) sesso : varchar(30) compleanno : date u_status : varchar(30) u_posts : varchar(30) descrizione : varchar(100) relazioni : varchar(30) user_cover : varchar(30) user_image : varchar(30) user_reg_date : date recovery_account : varchar(30) facolta : varchar(20) grado : varchar(20)	social_network mebri_canali id_m : int(11) id_studente : int(11) id_canale : int(11)	social_network commenti_canali id_commento : int(11) post_id : int(11) user_id : int(11) contenuto : varchar(255) data : date
	social_network comments com_id : int(11) post_id : int(11) user_id : int(11) comment : varchar(255) comment_author : varchar(255) date : date	social_network post_canali id_post : int(11) id_canale : int(11) titolo : varchar(50) contenuto : varchar(255) post_img : varchar(255) date : date
	social_network professori id_prof : int(11) nome : varchar(20) cognome : varchar(20) password : varchar(20) corso : varchar(30) data : date email : varchar(30)	social_network posts post_id : int(11) user_id : int(11) post_content : varchar(200) upload_img : varchar(255) post_date : date

Figura 18. Figura che mostra il Database finora

2.3. CREAZIONE PROFESSORI / GESTIONE DEGLI INSCRITTI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Creazione Professore con campi errati;	✗
Creazione Professore con campi corretti;	✓
Premere il tasto “Crea” senza aver inserito nulla;	✗
Premere il tasto “Crea” inserendo i campi correttamente;	✓
Provare a creare un Professore (rispettando i requisiti) con Server nello stato di: Connesso;	✓
Provare a creare un Professore (rispettando i requisiti) con Server nello stato di: Down.	✗

CAPITOLO 4: RELAZIONI

1.1. RICHIESTE DI AMICIZIA: MODELLAZIONE

In questo capitolo analizzeremo gli aspetti che contraddistinguono il concetto di relazioni.

Ogni studente mediante l'apposito menù in alto potrà accedere alla pagina Relazioni, in essa verranno visualizzati l'elenco delle persone con cui l'utente in questione ha stretto delle amicizie.

La fase che precede la stretta di amicizia è chiamata "Richiesta di amicizia", la quale può essere inoltrata sia dal Mittente che dal Ricevente, ovvero non è importante chi istaura per primo una richiesta.

Dopo che il Mittente ha inviato una richiesta di amicizia potrà verificare tutte le richieste che attendono una risposta dall'apposito menù "richieste di amicizie da te effettuate", le quali potranno essere cancellate in caso di ripensamento.

Il Ricevente nella sezione "richieste di amicizia ricevute", vedrà notificata una richiesta di amicizia e potrà scegliere se accettarla (e quindi stringere amicizia) oppure eliminarla.

Inoltre anche dopo l'istaurazione dell'amicizia entrambi i membri potranno cancellarla in qualsiasi momento.

1.2. RICHIESTE DI AMICIZIA: IMPLEMENTAZIONE

I Tuoi Amici

Richieste di amicizia in sospeso

frescobaldo
Libero
Ingegneria - frescolibero@gmail.com
- Italia - Maschio
Registrato il: 2019-06-11

Cancella Dagli Amici

Rosalba Rossa
Ingegneria - rosalba@gmail.com -
Italia - Femmina
Registrato il: 2019-06-24

Cancella Dagli Amici

Richiesete di amicizia da te effettuate

frescobaldo
Libero
 Ingegneria - frescolibero@gmail.com
 - Italia - Maschio
 Registrato il: 2019-06-11

Elimina richiesta di amicizia

Richieste di amicizia ricevute

Pierpaolo
Gumina
 Ingegneria - pier@gmail.com - Italia - Maschio
 Registrato il: 2019-06-11

Elimina richiesta di amicizia
 Accetta richiesta di amicizia

1.3. RICHIESTE DI AMICIZIA: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Molti bug iniziali specialmente per l'estrapolazione dei dati;	✗
Provare ad aggiungere o rimuovere richieste con Server nello stato di: Connesso;	✓
Provare ad aggiungere o rimuovere richieste con Server nello stato di: Down.	✗

2.1. SUGGERIMENTO DI AMICIZIA: MODELLAZIONE

Il principio di funzionamento dei suggerimenti di amicizia, in piattaforme come Facebook, è realizzato tramite un algoritmo che analizza centinaia di parametri differenti, ognuno con “peso” differente.

Probabilmente per suggerire nuove amicizie ci sono degli amici in comune che si hanno con altri utenti del social network medesimo. In pratica, vedendo che due utenti hanno diversi amici in comune, giunge alla conclusione che quest’ultimi possano conoscersi.

Tra le molteplici informazioni che si potrebbero analizzare, ci sono anche la città di origine, la città in cui si vive attualmente, la scuola frequentata, il lavoro, le relazioni familiari, i luoghi in cui ci si è registrati oppure sfruttare l’intelligenza artificiale unita ai milioni e milioni di informazioni di cui dispongono per fare in modo che i suggerimenti siano quanto più accurati possibili.

Nel mio progetto realizzo una dimostrazione di come possa funzionare un simile sistema.

Come citato sopra, la domanda su cui si basa il suggerimento di amicizia è essenzialmente una:

“cosa abbiamo in comune?”

Avere delle informazioni in comune permette di far in modo che una persona compaia o meno nella lista. In particolare le informazioni che sfrutto nel progetto sono due: anno di nascita di ciascun studente e facoltà di appartenenza. Con più informazioni a disposizione è possibile estendere l’algoritmo ulteriormente con altre informazioni così da renderlo più accurato.



2.2. SUGGERIMENTO DI AMICIZIA: IMPLEMENTAZIONE

Come citato nel paragrafo precedente, l'algoritmo di suggerimento di amicizia opera sfruttando le similitudini tra i vari utenti della piattaforma. L'algoritmo cerca di suggerire nuove amicizie tramite due campi:

1. La facoltà di appartenenza;
2. Stesso anno di nascita.

Ciò permette di categorizzare gli utenti in modo tale che i suggerimenti siano quanto più precisi possibili.

L'algoritmo si basa su 4 diverse query al database, in particolare:

1. Prima query: ha lo scopo di estrapolare le informazioni relative all'utente che ha effettuato il login, come id_utente, tipo di facoltà', anno di nascita, ecc...
2. Seconda query: una volta estrapolate le informazioni dell'utente loggato possiamo inserirle nella seconda query che darà come risultato una lista di utenti con anno di nascita in comune e che frequentano la stessa facoltà universitaria;
3. Terza e Quarta query: ottenuta la lista, quindi, andranno tolti eventuali utenti con cui si è stretta amicizia precedentemente o che si sia inviata la richiesta (o ricevuta) ed è in attesa di essere valutata. Le query verificano che ogni utente della lista sia presente nella tabella relazione, se sono presenti allora l'utente andrà scartato altrimenti diventerà un suggerimento di amicizia valido.

Suggerimenti di Amicizia (BETA)

**Pierpaolo
Gumina**

Ingegneria - pier@gmail.com - Italia -
Maschio

Registrato il: 2019-06-11

Invia richiesta di amicizia

**frescobaldo
Libero**

Ingegneria - frescolibero@gmail.com
- Italia - Maschio

Registrato il: 2019-06-11

Invia richiesta di amicizia

2.3. SUGGERIMENTI DI AMICIZIA: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Molti bug iniziali specialmente per l'estrapolazione dei dati;	✗
Provare ad inviare richieste di amicizia con Server nello stato di: Connesso;	✓
Provare ad inviare richieste di amicizia con Server nello stato di: Down.	✗

CAPITOLO 5: COMPRAVENDITA LIBRI, BOOKCROSSING E CONDIVISIONE DEGLI APPUNTI

1.1. COMPRAVENDITA LIBRI: MODELLAZIONE

Prima di analizzare nel dettaglio la fase di modellazione è opportuno dare uno sguardo ai Casi d'uso.

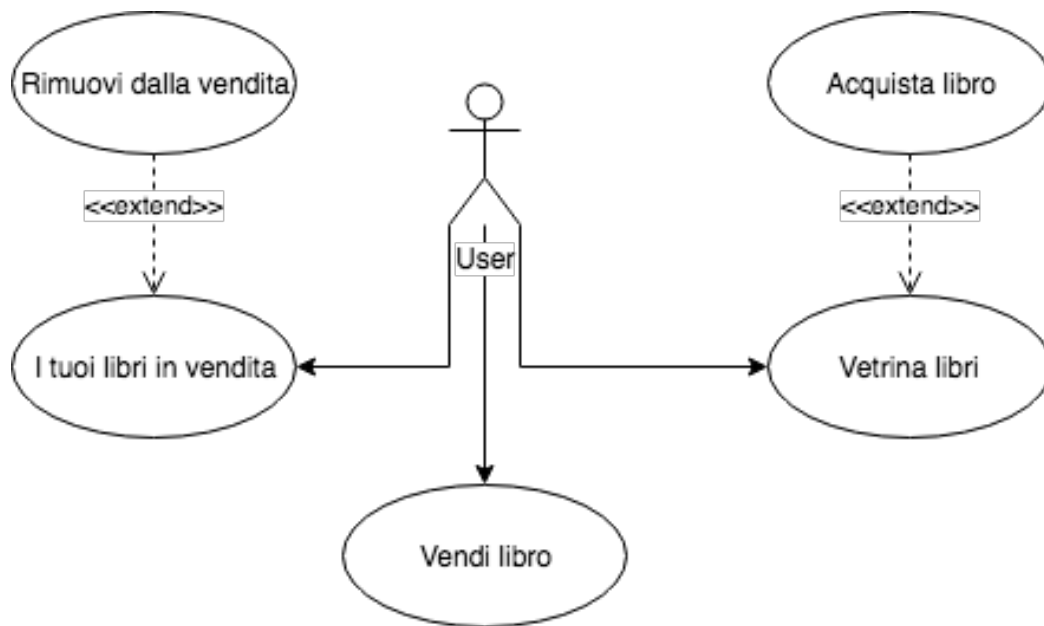


Diagramma dei casi d'uso

Nella piattaforma è stata aggiunta la compravendita di libri usati, acquistabili tramite carta di credito/prepagata.

Uno studente per poter vendere un libro in suo possesso dovrà recarsi nell'apposito menù "Vendi libro", dopo aver inserito alcune informazioni, il libro sarà pronto per essere acquistato.

L'utente inoltre ha la possibilità di gestire tutti i propri libri messi in vendita. Recandosi nel menù "I tuoi libri in vendita" può decidere in qualsiasi momento di rimuovere i libri dalla vendita (magari perché ha trovato un possibile acquirente di persona).

Analizziamo nel dettaglio i passi che portano dalla messa in vendita di un libro all'acquisto.

1.2. COMPRAVENDITA LIBRI: IMPLEMENTAZIONE

Da come è possibile notare dall'immagine per poter vendere un libro sono richieste le seguenti informazioni: Nome libro, autore, edizione, condizioni, prezzo, corso, facoltà, un contatto telefonico ed una foto del libro.

Le prime 3 informazioni riguardano il libro in maniera oggettiva, mentre le condizioni e il prezzo sono informazioni soggettive inserite dal venditore.


Sarebbe opportuno che quest'ultime siano quanto più oggettive possibile, in modo da favorire una transazione onesta e corretta. In questa fase è possibile inserire una foto del libro dare prova delle effettive condizioni inserite dal venditore.

Le condizioni predisposte consistono in 4 tipologie: pessime, discrete, ottime, quasi nuovo.


Compilati tutti i campi basterà premere il tasto "Metti in vendita" per confermare e rendere acquistabile da tutti gli utenti nella sezione "Vetrina libri".

Vendita libro


Inserisci le informazioni del libro




Nome Libro




Autore




Edizione




Pessime




Prezzo in €



Ingegneria



Inserisci un tuo contatto telefonico



Scegli file


Nessun file selezionato

Metti in Vendita

Vendi

Vetrina Libri

I tuoi libri in vendita




Sistemi operativi. Concetti ed esempi

Silberschatz, Peter Baer Galvin, Greg Gagne - 9

Prezzo € 20

Vedi più informazioni Acquista



Basi di dati. Con Connect

Paolo Atzeni, Stefano Ceri, Piero Fraternali, Stefano Paraboschi, Riccardo Torlone - 5

Prezzo € 23

Vedi più informazioni Acquista

Analizziamo adesso la fase di acquisto.


La fase di acquisto è analoga a qualsiasi sistema di pagamento effettuato tramite PayPal e simili. In particolare nella mia piattaforma sfrutto il sistema di pagamento tramite Stripe: grazie alle API messe a disposizione ho potuto incorporare strumenti di pagamento rapidi e sicuri.

Premendo su “Acquista” viene intrapresa la procedura di pagamento che consiste nei seguenti passi:

- 1) Per interfacciarsi con le API di Stripe è necessario creare una Sessione contenente una ApiKey (fornita dopo la registrazione nel sito stripe.com);
- 2) Successivamente deve essere composta la stringa \$session, nella quale inseriremo le informazioni di acquisto quali: nome, prezzo, modalità di pagamento, valuta, ecc...
- 3) Se i dati di pagamento inseriti sono corretti la transazione andrà a buon fine e l'utente sarà reindirizzato alla pagina di “Success Payment”, se invece i dati inseriti non sono corretti verrà reindirizzato alla pagina di “Cancel Payment”.

Una volta completato il pagamento il libro in questione verrà rimosso dalla Vetrina libri.


In qualsiasi momento è possibile rimuovere dalla Vetrina libri i propri volumi messi in vendita recandosi nella sezione “I tuoi libri in vendita” e premendo il tasto “Rimuovi dalla vetrina”.


account_prova
TEST

Nome: Sistemi operativi. Concetti ed esempi.
Autore: Silberschatz, Peter Baer Galvin, Greg Gagne. Edizione: 9

20,00 €

Corso di: Ingegneria




Powered by  | [Condizioni](#) [Privacy](#)

Paga con carta

Email


Informazioni sulla carta

1234 1234 1234 1234

MM / AA

CVC



Titolare della carta

Paese o regione

Italia

▼

Paga 20,00 €

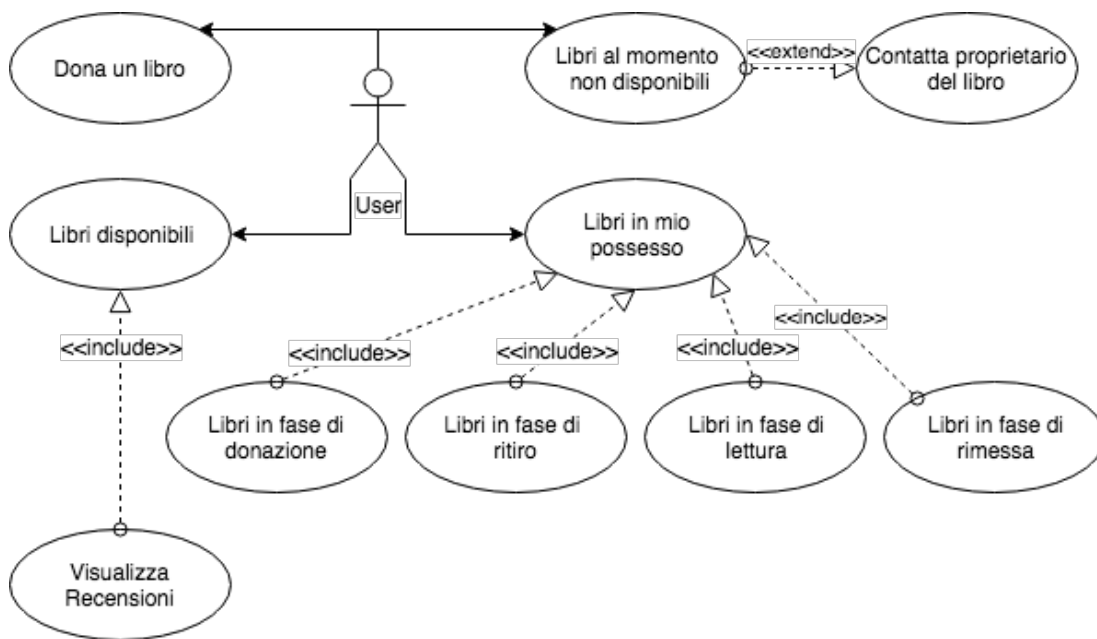
1.3. COMPRAVENDITA LIBRI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Inserimento di dati scorretti o mancanti nella fase di messa in vendita;	✗
Inserimento di dati nella fase di messa in vendita con Server nello stato di: Connesso.	✓
Inserimento di dati nella fase di messa in vendita con Server nello stato di: Down.	✗

Sono seguiti molte verifiche di natura non sperimentale le quali hanno portato alla risoluzione di molteplici bug (come test di pagamento, ecc.).

2.1. BOOKCROSSING: MODELLAZIONE



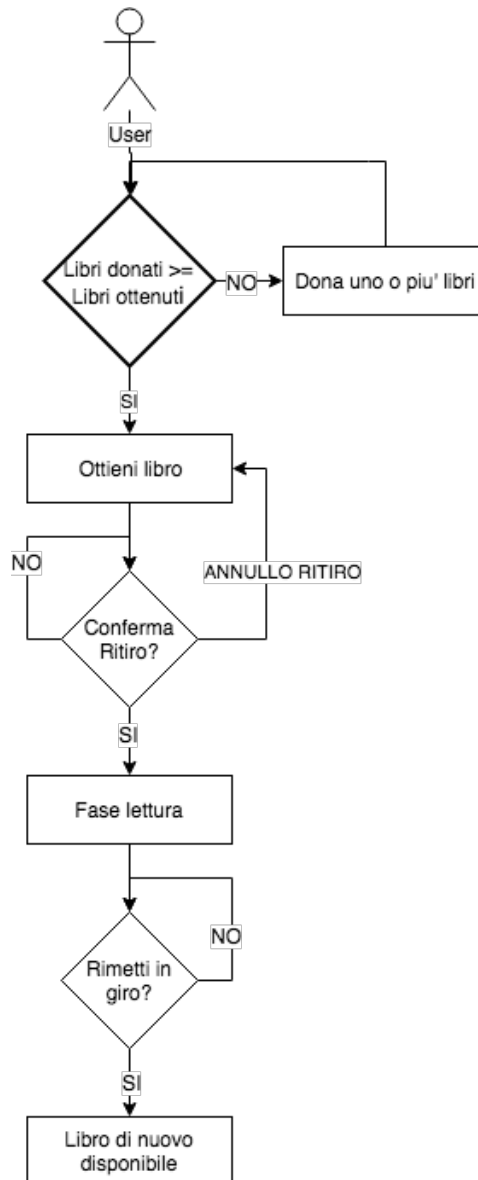
Il BookCrossing ha un nome composto da due termini in lingua inglese: book (libro) e crossing (attraversamento, passaggio). Si tratta di una iniziativa di distribuzione gratuita di libri che legano la passione per la lettura e per i libri alla passione per la condivisione delle risorse e delle conoscenze.

L'idea di base è di lasciare dei libri nell'ambiente naturale, compreso quello urbano o "into the wild" ovvero dovunque si preferisca affinché possano essere ritrovati e quindi letti da altri che eventualmente possono commentarli oppure farli direttamente proseguire nel loro viaggio.

Il meccanismo del Bookcrossing ruota intorno all'esistenza di un sito web attraverso il quale è possibile scambiarsi dei libri, rispettando alcune regole fondamentali:

- 1) Il buon senso è sicuramente un principio fondamentale del bookcrossing: non bisogna dare informazioni errate, non bisogna danneggiare i libri e non bisogna rubarli o tenerli per troppo tempo;
- 2) Ogni nuovo utente per ottenere un libro deve prima donarne uno poiché se nessuno donasse libri non c'è ne sarebbero in circolazione. Ecco che una regola impone che il numero dei libri donati sia maggiore o uguale al numero dei libri in proprio possesso, così da fomentare la circolazione di più libri e quindi il successo del meccanismo del bookcrossing.

2.2. BOOKCROSSING: IMPLEMENTAZIONE



Per poter realizzare il bookcrossing è necessario, come abbiamo affermato, un sito web che tenga traccia delle informazioni di ciascun libro.

Analizziamo un ciclo completo di stadi di un libro:

- Ogni libro deve superare lo stadio della donazione recandosi nel menù “Bookcrossing → Dona”, così ogni utente inserisce in questa sezione le informazioni del libro quali: nome libro, autore, una immagine illustrativa ed il luogo corrispondente dove verrà depositato il libro donato;
- Il libro non sarà reso subito disponibile, infatti quest’ultimo verrà inserito in una sezione preliminare denominata “Libri in fase di donazione”. In questa fase l’utente può decidere di cancellare la donazione o confermarla, se l’utente conferma la donazione significa che avrà depositato il libro nel luogo indicato precedentemente;

- Dopo la conferma il libro è disponibile a tutti i membri della piattaforma. Se un utente fosse interessato ad un libro tra quelli disponibili dovrà recarsi nel menù “Bookcrossing → Libri disponibili” e premere il tasto “Ottieni” così il libro in questione sarà immediatamente reso non disponibile agli altri utenti ed inserito nel menù “Libri in fase di ritiro”;
- Il libro, come per la fase di donazione, risulterà inserito in uno stadio preliminare denominato “Libri in fase di ritiro”. In questa fase l’utente avrà la possibilità di cancellare il ritiro del libro, ripristinando la disponibilità di quest’ultimo o confermarne il ritiro così da prelevare il libro nel luogo indicato;
- Confermato il ritiro del libro, quest’ultimo risulterà nello stadio “libri in fase di lettura”. Durante la fase di lettura, qualora un utente impiegasse troppo tempo per la lettura del libro è possibile contattarlo, tramite chat privata, per sollecitarlo a concludere la lettura per rimettere nuovamente il libro in circolazione e renderlo così disponibile alla comunità di lettori c.d. “liberi”;
- Terminata la lettura premendo il tasto “metti in giro” comparirà un pop-up che darà all’utente la possibilità di recensire il libro, tale recensione sarà visibile a tutti gli utenti. Non è obbligatorio lasciare delle recensioni (non scrivendo nulla nell’apposita sezione dedicata alle recensioni). Alla fine del procedimento il libro si troverà nello stadio preliminare denominato “libri in fase di rimessa”;
- L’ultimo stadio prevede la rimessa in giro del libro rendendolo così di nuovo fruibile alla comunità. Premendo il tasto “Conferma Rimessa” comparirà un pop-up che darà all’utente la possibilità di inserire il luogo dove il libro è stato già depositato. Alla fine il libro sarà di nuovo reso disponibile nel menù “Libri disponibili”, pronto per essere letto da un altro utente.

♥ Dona Libro ♥

Inserisci le informazioni del libro

Nome Libro

Autore

Scegli file

Nessun file selezionato

Luogo

Metti in Giro

Il DB (data base) viene così aggiornato con le modifiche su esposte.

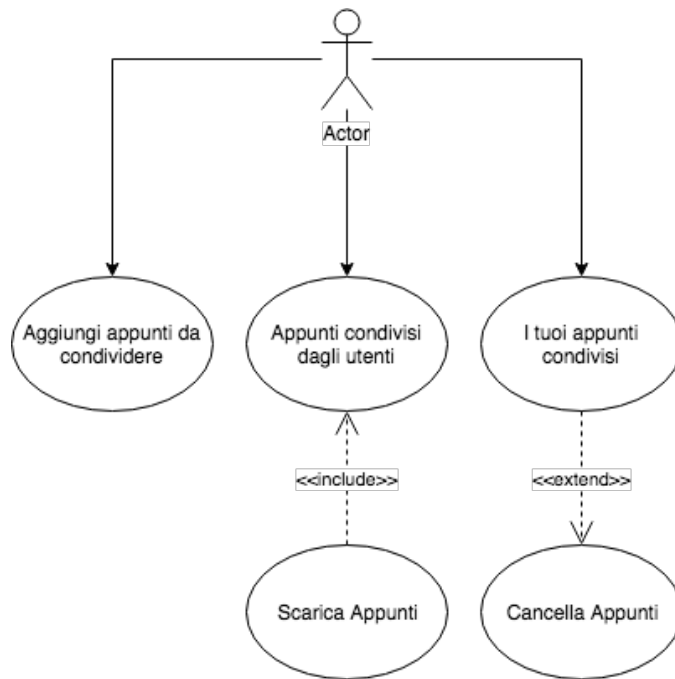
36

2.3. BOOKCROSSING: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Inserimento di dati scorretti o mancanti nella fase di donazione;	✗
Inserimento di dati nella fase di donazione con Server nello stato di: Connesso.	✓
Inserimento di dati nella fase di donazione con Server nello stato di: Down.	✗

3.1. CONDIVISIONE APPUNTI: MODELLAZIONE



La condivisione di appunti è un meccanismo che permette lo scambio di appunti tra studenti.

Il suddetto meccanismo consiste in due passi:

- Creazione dell'elemento "Appunti";
- Ottenimento, tramite Link Web, del materiale reso disponibile dall'utente.

Lo studente che rende pubblici i propri appunti può in qualsiasi momento eliminarli.

3.2. CONDIVISIONE APPUNTI: IMPLEMENTAZIONE

L'utente per condividere i propri appunti dovrà recarsi nel menù "Condivisione Appunti → Condividi". In questa sezione verranno chiesti all'utente alcune informazioni riguardanti gli appunti da condividere quali: link, materia, corso e descrizione (quest'ultima facoltativa).

Una volta premuto il tasto "Condividi Appunti", gli appunti saranno disponibili al download da tutti gli utenti.

Le piattaforme di sharing consigliate sono Google Drive e Dropbox, ma è possibile inserire qualsiasi servizio di file-sharing disponibili sul mercato.

Nel caso in cui uno studente volesse eliminare alcuni suoi appunti si dovrà recare nel menù "I tuoi appunti condivisi" e premere il tasto "Cancella Condivisione".

Condividi Appunti

Inserisci le informazioni necessarie

	<input type="text" value="Link"/>
	<input type="text" value="Materia"/>
	<input type="text" value="Corso"/>
	<input type="text" value="Descrizione"/>

Condividi Appunti

Condividi +

Appunti condivisi dagli utenti

I tuoi appunti condivisi

Cerca tra gli appunti

Cerca

Materia: Sistemi Operativi

Scarica

Corso: Ingegneria

Caricato da: Pierpaolo
Gumina il: 2019-07-20

Descrizione: Qui troverete tutti i miei appunti di SO. Ciao!

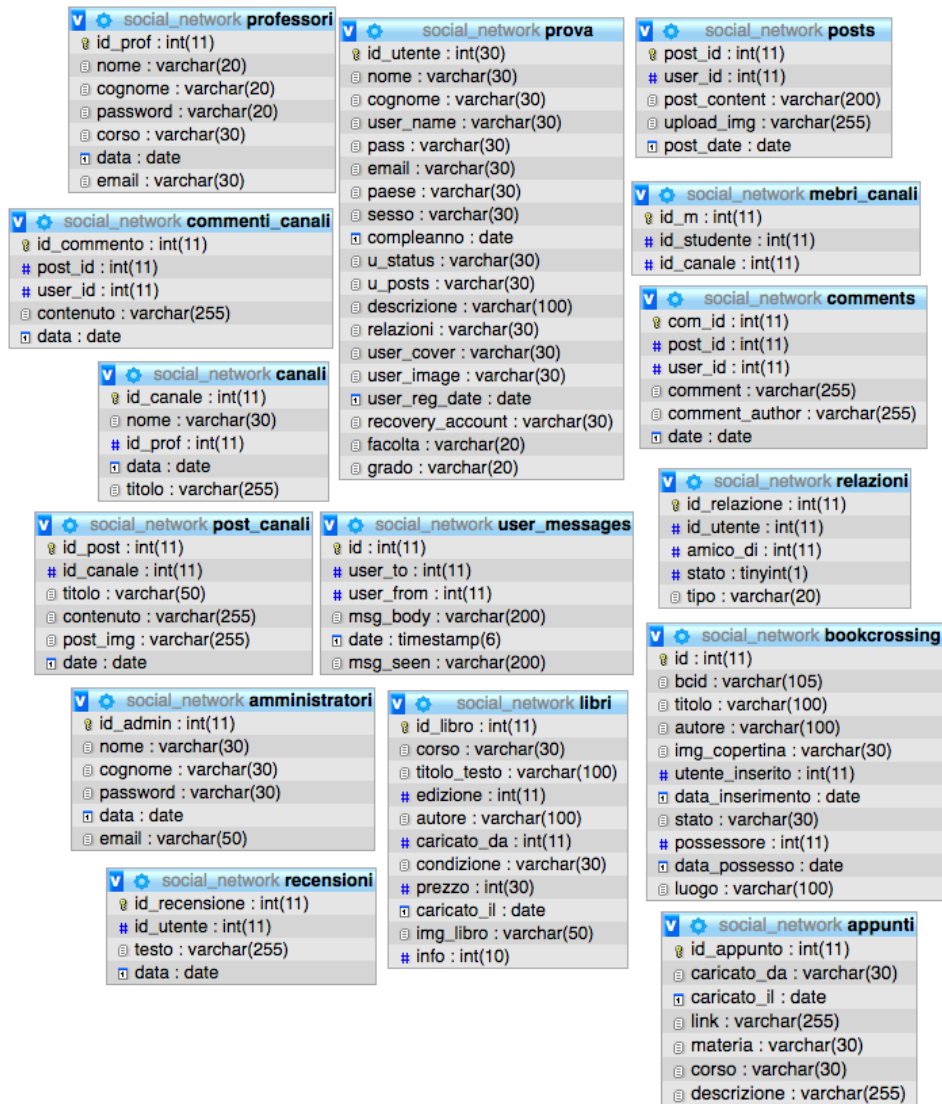
Materia: Programmazione III

Scarica

Corso: Scienze Informatiche

Caricato da: Pierpaolo
Gumina il: 2019-07-20

Descrizione: Qui troverete tutti i miei appunti di Programmazione III. Ciao!



3.3. CONDIVISIONE APPUNTI: TEST

Conclusasi la fase di Implementazione sono seguite varie fasi di Testing, per accertarsi che il sistema funzioni correttamente. Nella tabella di seguito sono presenti i vari test effettuati con il relativo esito. I test sono stati effettuati tenendo conto dei requisiti del sistema.

Tipo di Test Effettuato	Esito
Inserimento di dati scorretti o mancanti nella fase di condivisione;	✗
Inserimento di dati nella fase di condivisione con Server nello stato di: Connesso.	✓
Inserimento di dati nella fase di condivisione con Server nello stato di: Down.	✗