



A.D. 1308
unipg

UNIVERSITÀ DEGLI STUDI
DI PERUGIA

Ricerca e prenotazione posto Lezione tramite bot Telegram

2021/2022

Christian Angileri [330925]
Alessio Cassieri [324396]

Ingegneria del Software
Docente: Alfredo Milani

Indice:

1. Abstract Summary	3
2. Obiettivo del Progetto	3
3. Analisi dei requisiti	4
3.1 Dominio applicativo	4
3.2 Glossario	4
3.3 Requisiti funzionali	4
3.4 Requisiti non funzionali	5
4. Diagramma	6
a. Di casi d'uso	6
b. Di classe	7
c. Di sequenza	8
d. Di stato	8
e. Di attività	9
5. Test funzionali	10
6. Realizzazione e implementazione	11
7. Demo progetto	13
8. Collegamenti esterni	18

1 Abstract Summary

Il progetto consiste nella realizzazione di un bot Telegram per la ricerca e prenotazione di un posto in aula da parte di uno studente dell'università degli studi di Perugia.

Il progetto è sviluppato tramite Python con supporto della libreria 'pyTelegramBotAPI', utilizziamo anche un database di appoggio per l'identificazione degli utenti e per contenere i dati riguardo i vari corsi, orari e professori, realizzato tramite SQLiteStudio.

2 Obiettivi Progetto

Il primo punto del progetto è garantire identificazione dell'utente, in modo da far accedere soltanto Studenti registrati, tramite inserimento della matricola e password.

Il bot deve garantire la riservatezza delle credenziali, almeno per quanto riguarda la password, dato che lo storico chat rimane all'interno di Telegram. Infatti appena viene inserita la password, il bot si assicura di eliminare il messaggio dell'utente, pure se la password inserita non è corretta.

Effettuato l'accesso, lo Studente, seguendo le istruzioni, potrà prenotarsi un posto per una certa lezione.

Lo Studente visualizzerà informazioni aggiuntive su una lezione prima della prenotazione, come:

- posti disponibili;
- docente;
- aula;
- orario.

Si può visualizzare la lista delle Lezioni prenotate e, nel caso lo Studente decida di non partecipare ad una lezione, può cancellare la prenotazione.

Lo Studente può effettuare il logout dal suo dispositivo.

3 Analisi dei Requisiti

3.1 Dominio applicativo:

- Studenti dell'università degli studi di Perugia

3.2 Glossario:

- User:
 - coloro che tentano di autenticarsi al sistema
- Bot:
 - Utenti artificiali programmati per interagire con utenti reali per effettuare determinate operazioni.
- Markup:
 - modulo che fornisce un'interfaccia unificata che permette, attraverso adeguata implementazione su python, di inserire i vari pulsanti presenti sui bot di telegram
- Raspberry:
 - Computer a scheda singola progettata per ospitare sistemi operativi basati sul kernel linux. Nel caso del progetto permette di tenere attivo il bot in modo tale da essere utilizzato in qualsiasi momento per la prenotazione.

3.3 Requisiti funzionali:

- **Autenticazione**

La funzione nasce per l'esigenza di riconoscere lo studente e per evitare truffe. È necessario che si possano controllare i posti occupati e i rispettivi studenti per una migliore gestione delle politiche di sicurezza del covid-19.

- **Scelta anno e corso**

Saranno presenti dei markup che faranno scegliere all'utente, cliccando nel pulsante interessato, l'anno dei corsi che vuole seguire.

Successivamente verranno visualizzate tutte le lezioni e gli orari della settimana.

- **Prenotazione e cancellazione posto**

Saranno presenti ulteriori markup che permetteranno all'utente la scelta del posto. Una volta scelto non sarà possibile ad altri utenti scegliere il medesimo posto. È possibile cancellare la propria prenotazione in modo da permetterla ad altri studenti.

3.4 Requisiti non funzionali:

- **Di prodotto**

Il bot dovrà essere disponibile in qualsiasi momento per permettere la prenotazione in presenza.

- **Organizzativi**

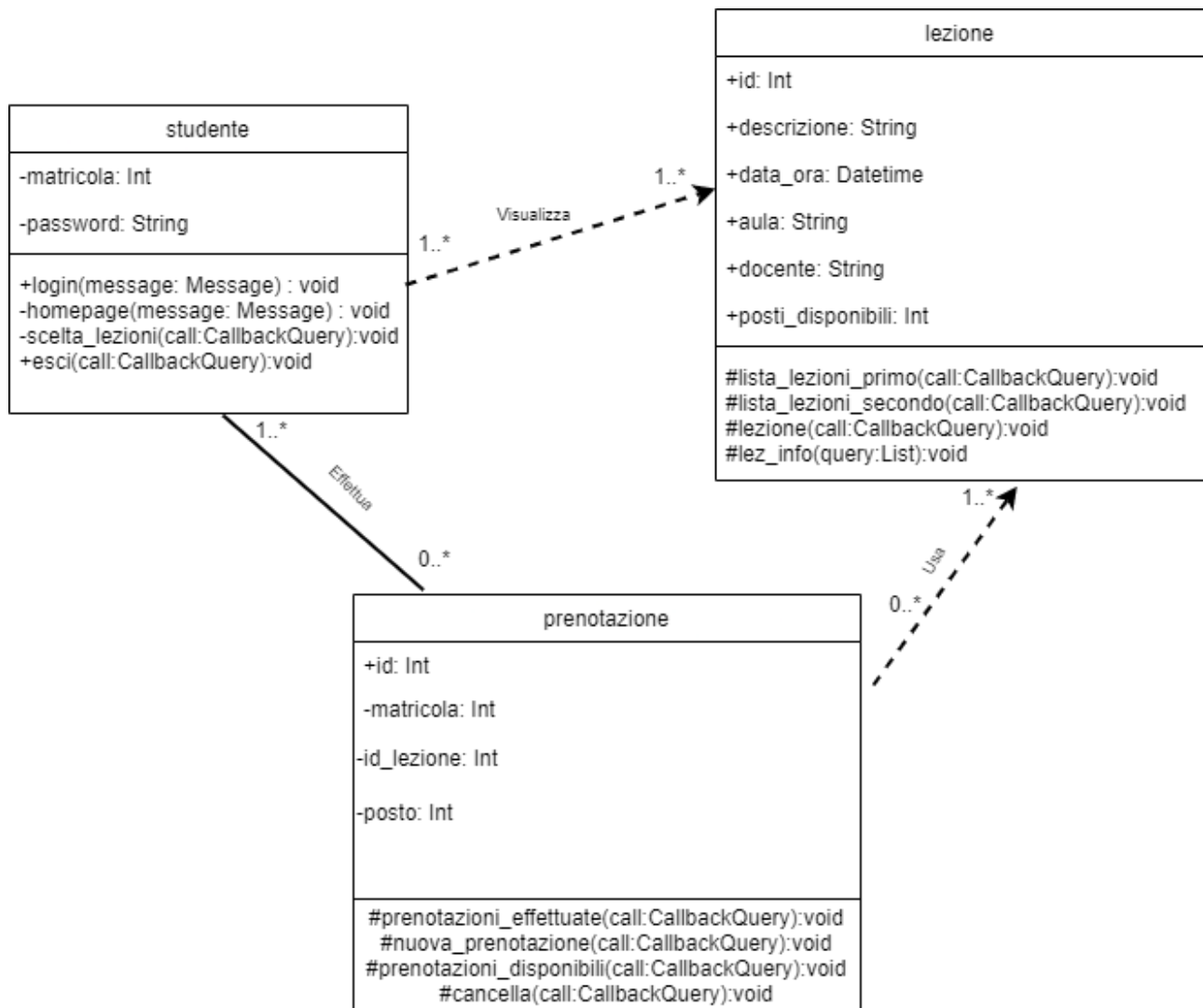
È necessario che gli utenti siano in possesso dell'applicazione Telegram, del link del bot e dei dati identificativi per poter effettuare la prenotazione.

- **Esterni**

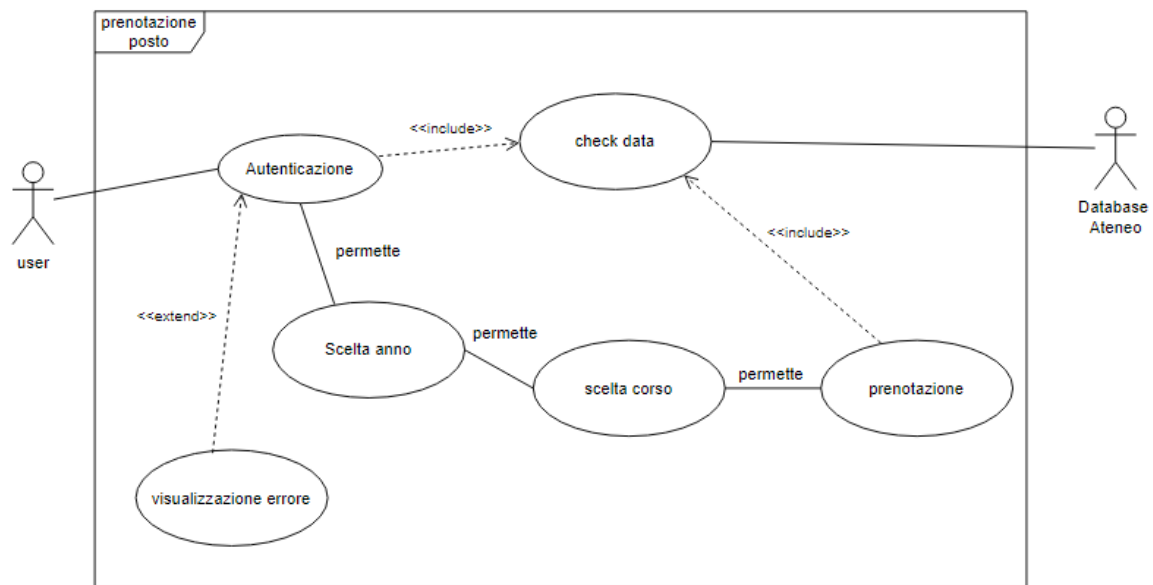
Il bot si collegherà al database dell'ateneo che contiene tutti i dati necessari all'autenticazione.

4 Diagramma

a. Di classe



b. Di casi d'uso



Caso d'uso: UC1 Prenotazione posto

Attori: Utente

Precondizioni: L'utente non è autenticato presso il sistema

Postcondizioni: utente, iscritto unipg, può accedere alla prenotazione al posto aula attraverso numero matricola e password dell'Ateneo

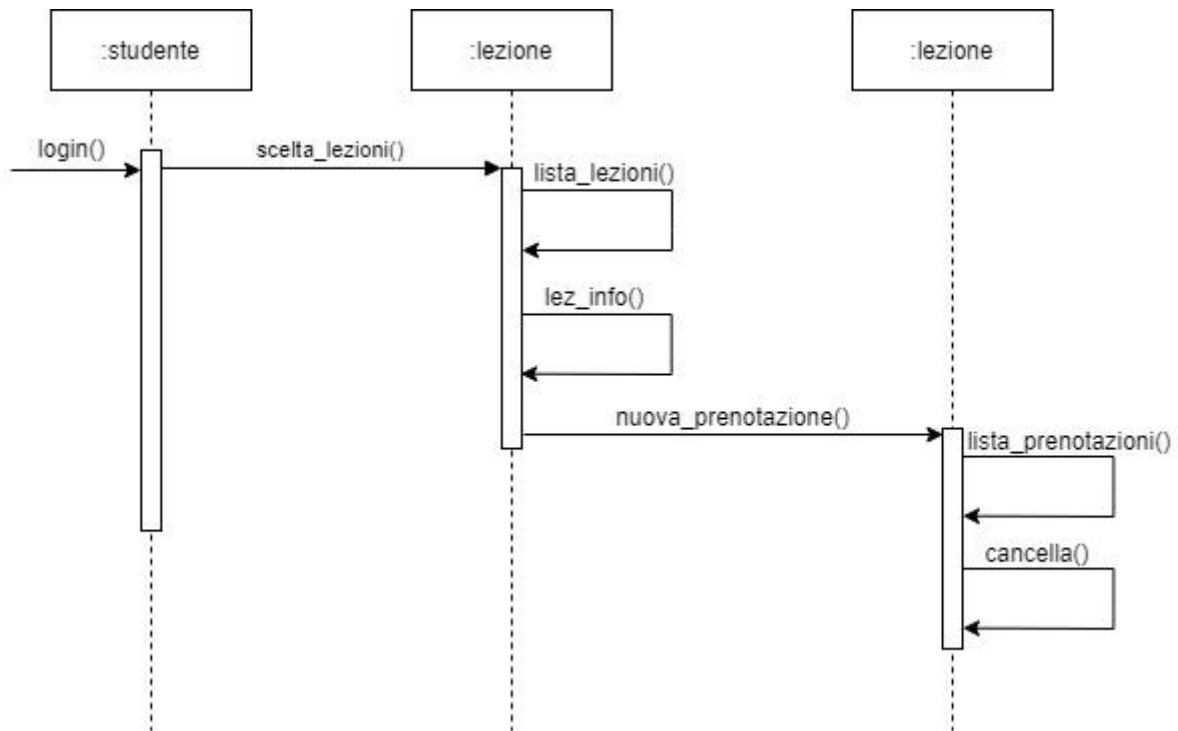
Scenario principale:

1. Utente si collega al bot di Telegram
2. Inizia conversazione e vengono inseriti dati identificativi
3. Utente sceglie anno corso
4. Utente sceglie corso da seguire
5. Utente sceglie posto in aula

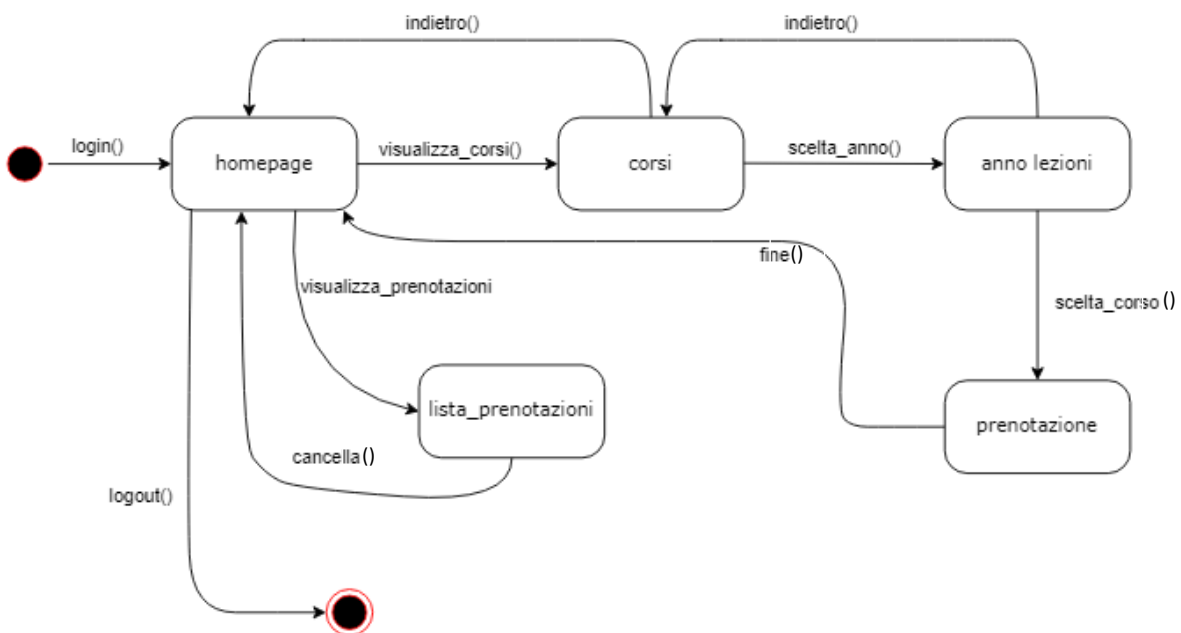
Scenario secondario:

1. Nel caso vengano inseriti dati identificativi non validi, viene richiesto di inserire nuovamente i dati.
2. Nel caso vengano inseriti degli input da tastiera (nel momento in cui è necessario premere i pulsanti a schermo) viene mandato un messaggio di errore.

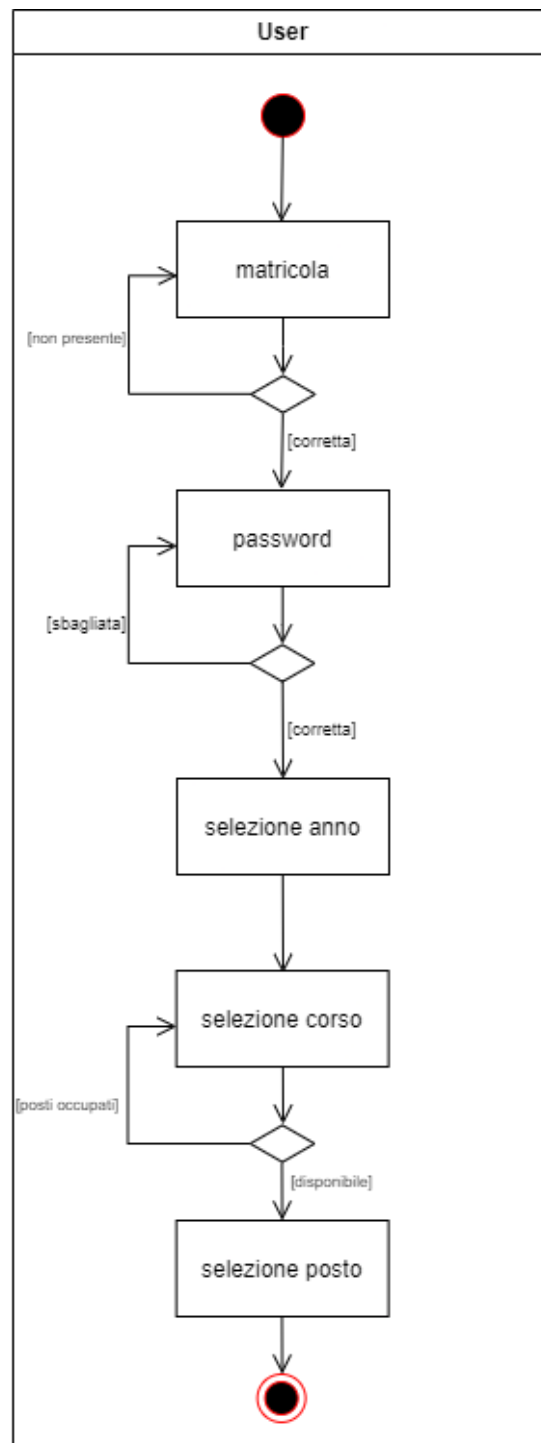
c. Di sequenza



d. Di stato



e. Di attività



5 Test funzionali

Il caso test analizza il login dello studente: numero matricola e password:

Parametri	Classe VALIDA	Classe NON VALIDA
Matricola	Len(Matricola) = 6 Isnumeric(Matricola) = True	Len(Matricola) != 6 Isnumeric(Matricola) != True
Password	6 <= Len>Password) <= 20 Almeno 1 numero Almeno 1 lettera maiuscola	Len>Password) < 6 or Len>Password) > 20 Nessun numero Nessuna lettera maiuscola

Parametri	Classe estrema VALIDA	Classe estrema NON VALIDA
Matricola	N/A N/A	N/A N/A
Password	Len>Password) == 6 or Len>Password) == 20 Almeno 1 numero Almeno 1 lettera maiuscola	Len>Password) == 5 or Len>Password) == 21 N/A N/A

Casi di test

Caso n.	n. matricola	Password	validità	motivazione
1	330925	Christian1	Valido	
2	324396	Alessio2	Valido	
3	345	Albero7	Non valido	Len(matricola) != 6
4	123abc	Carlo2	Non valido	Matricola non numeric
5	320145	abc	Non valido	Len>Password) < 6
6	330789	Supercalifragilistich26	Non valido	Len(password) > 20
7	325741	marco24	Non valido	No lettere maiuscole
8	333520	Giacomo	Non valido	No numeri

Casi test estremi

Caso n.	n. matricola	Password	validità	motivazione
1	300123	Guido1	Valido	Len(password) == 6
2	324385	Gianmarco.palazzolo1	Valido	Len(password) == 20
3	300124	Luca0	Non valido	Len(password) == 5
4	362589	Gianmarco.palazzolo20	Non valido	Len(password) == 21

6 Realizzazione e implementazione

Il bot è stato realizzato inizialmente su Telegram tramite @BotFather che permette la creazione e la gestione di canali. Successivamente il bot è stato programmato, come già detto prima, tramite Python con il supporto della libreria 'pyTelegramBotAPI' e della libreria 'sqlite3' per la connessione con il database, realizzato con SQLiteStudio, l'editor utilizzato è Atom. Per la gestione del database abbiamo utilizzato alcune query realizzate tramite SQL, per la creazione, rimozione e ricerca di campi.

Per permettere la continua disponibilità del bot, viene utilizzato un Raspberry Pi 3, dove al suo interno sono presenti:

- file .py che viene eseguito in automatico all'accensione;
- file .db (database).

Si può accedere al bot tramite un qualsiasi dispositivo con Telegram installato all'interno, pur avendo un account valido.

Il modello utilizzato è il Code&Fix, infatti il programma è testato ogni qualvolta viene completata una nuova funzione, e se si trovano errori, vengono corretti prima di passare avanti.

Esempi di alcune funzioni possono essere:

- nuova_prenotazione;
- cancella;
- lezione...

Per prima cosa il bot va cercato nella barra di ricerca apposita di Telegram, scrivendo @PrenotazionePostoBot , una volta trovato ti basterà seguire le indicazioni a schermo.

Finché non viene eseguito il comando '/start' il bot non entrerà in funzione.

Per eseguire l'accesso è necessario essere registrati all'ateneo, (per il nostro esempio abbiamo utilizzato le nostre matricole ed alcune di prova) ed inserire la matricola, che deve rispettare i seguenti criteri:

- Essere di 6 caratteri;

- Essere composta solo da numeri;

ed ovviamente la matricola deve corrispondere ad una all'interno del database. Una volta inserita la matricola correttamente viene chiesta la password, anche questa deve rispettare alcuni criteri:

- Deve avere tra i 6 e 20 caratteri compresi;
- Deve contenere almeno una lettera maiuscola;
- Deve contenere almeno 1 numero;

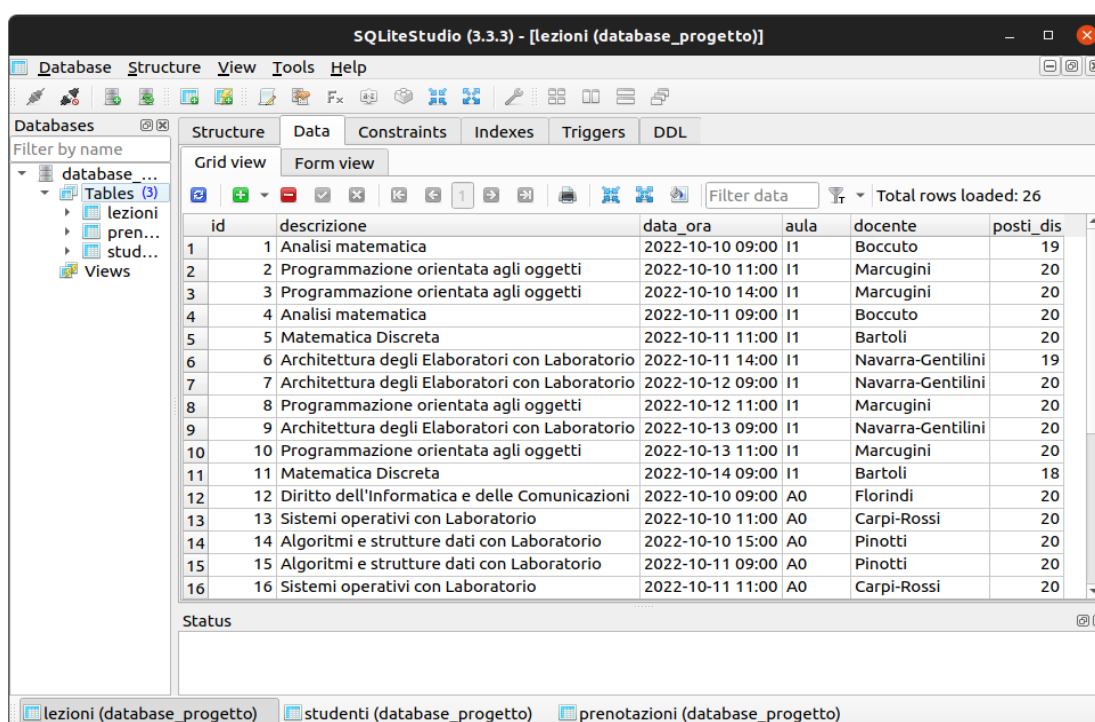
e comunque deve corrispondere alla password inserita nel database a quella rispettiva matricola.

Effettuato il login non sarà più necessario l'utilizzo della tastiera, ma basterà cliccare sui markup (bottoni) appostivi, seguendo le istruzioni.

La schermata principale è la homepage dove lo studente decide cosa visualizzare.

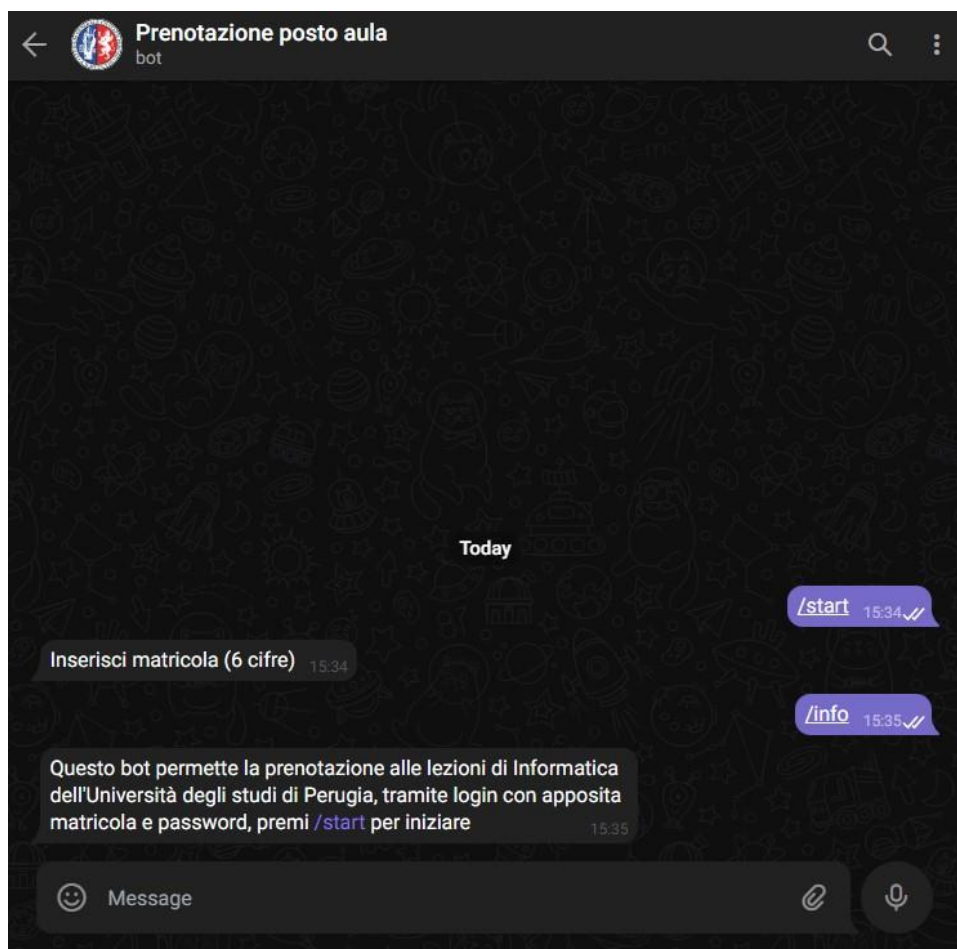
La prenotazione di una qualsiasi lezione può essere effettuata soltanto se lo studente non è già prenotato per la stessa (uno studente non può prenotare 2 posti per la stessa lezione), oppure se i posti disponibili, che sono 20, sono già tutti occupati.

Per quanto riguarda il database, sono state realizzate 3 tabelle che rispettano il diagramma di classe, di seguito una vista dei dati contenuti all'interno della tabella lezione:

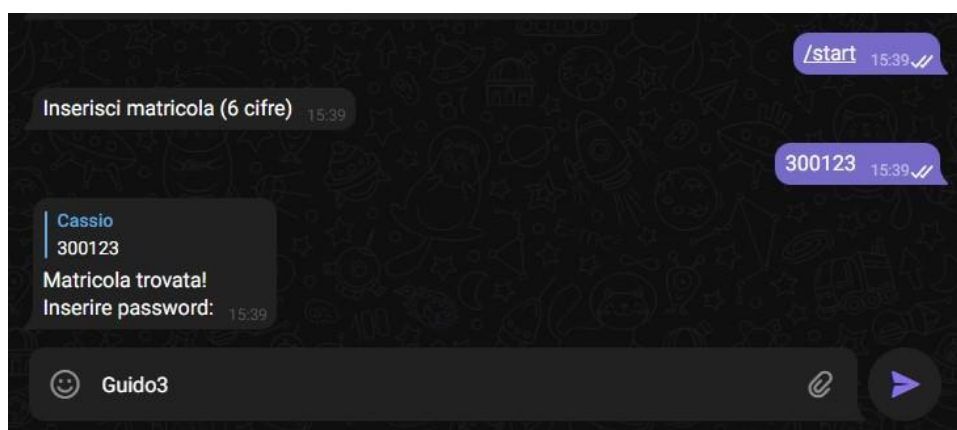


id	descrizione	data_ora	aula	docente	posti_dis
1	1 Analisi matematica	2022-10-10 09:00	I1	Boccuto	19
2	2 Programmazione orientata agli oggetti	2022-10-10 11:00	I1	Marcugini	20
3	3 Programmazione orientata agli oggetti	2022-10-10 14:00	I1	Marcugini	20
4	4 Analisi matematica	2022-10-11 09:00	I1	Boccuto	20
5	5 Matematica Discreta	2022-10-11 11:00	I1	Bartoli	20
6	6 Architettura degli Elaboratori con Laboratorio	2022-10-11 14:00	I1	Navarra-Gentilini	19
7	7 Architettura degli Elaboratori con Laboratorio	2022-10-12 09:00	I1	Navarra-Gentilini	20
8	8 Programmazione orientata agli oggetti	2022-10-12 11:00	I1	Marcugini	20
9	9 Architettura degli Elaboratori con Laboratorio	2022-10-13 09:00	I1	Navarra-Gentilini	20
10	10 Programmazione orientata agli oggetti	2022-10-13 11:00	I1	Marcugini	20
11	11 Matematica Discreta	2022-10-14 09:00	I1	Bartoli	18
12	12 Diritto dell'Informatica e delle Comunicazioni	2022-10-10 09:00	A0	Florindi	20
13	13 Sistemi operativi con Laboratorio	2022-10-10 11:00	A0	Carpi-Rossi	20
14	14 Algoritmi e strutture dati con Laboratorio	2022-10-10 15:00	A0	Pinotti	20
15	15 Algoritmi e strutture dati con Laboratorio	2022-10-11 09:00	A0	Pinotti	20
16	16 Sistemi operativi con Laboratorio	2022-10-11 11:00	A0	Carpi-Rossi	20

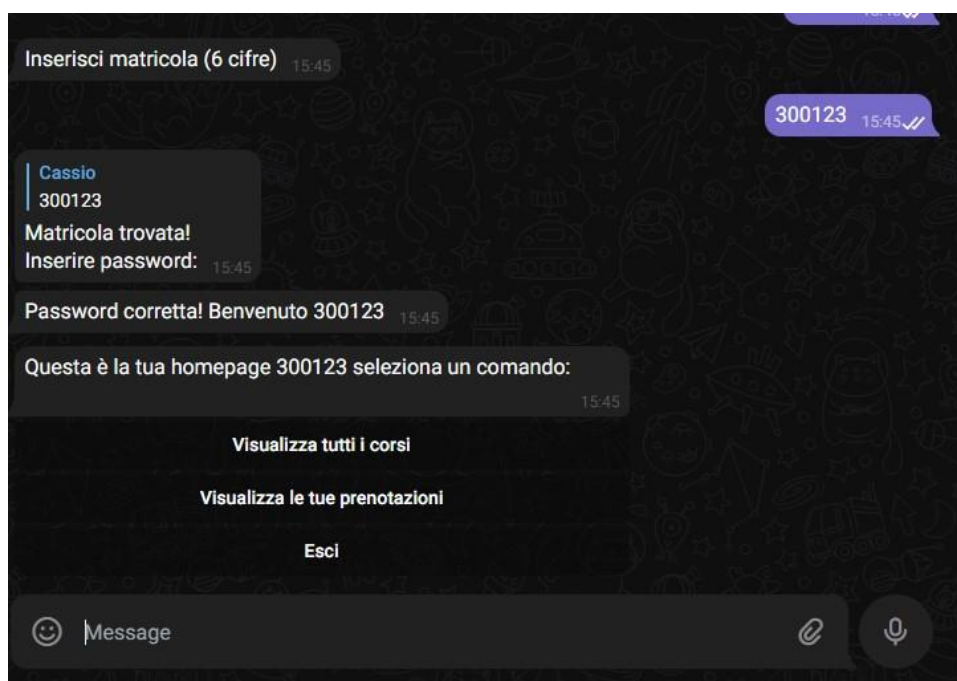
7 Demo progetto



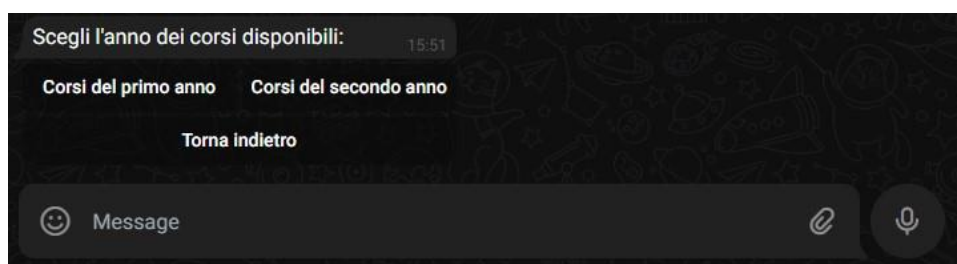
Schermata iniziale: vengono utilizzati i primi due comandi disponibili: '/start' e '/info'. Permettono, in ordine, di iniziare il login per la prenotazione e di essere a conoscenza di tutte le informazioni su cosa è possibile fare su questo bot



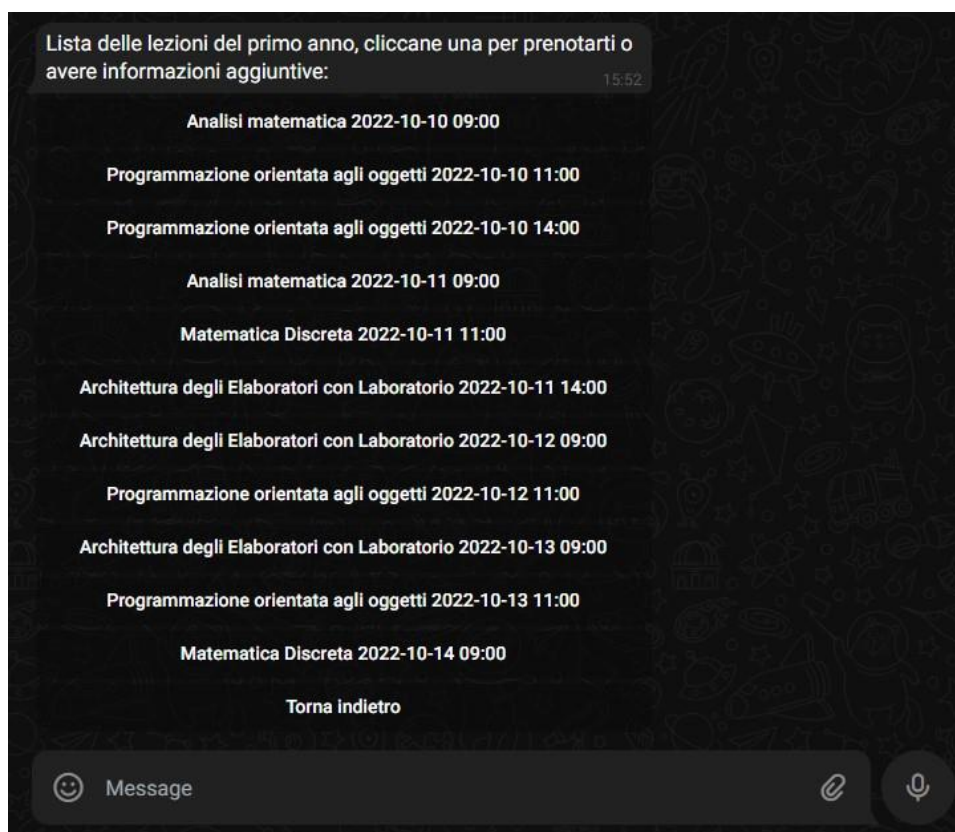
inserimento matricola: prima fase del login



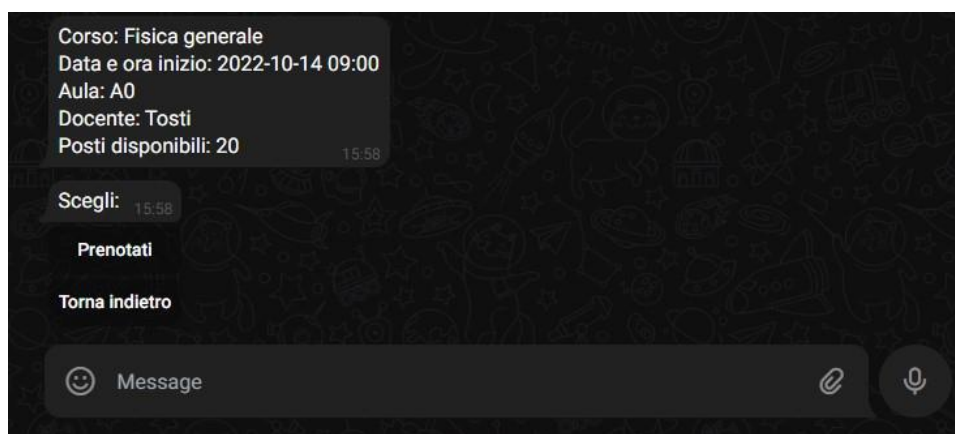
inserimento password e interfaccia homepage: dopo aver inserito la password, il login viene concluso, la password inserita cancellata e viene presentata la schermata di homepage che permette la scelta, tramite pulsanti, delle varie alternative.



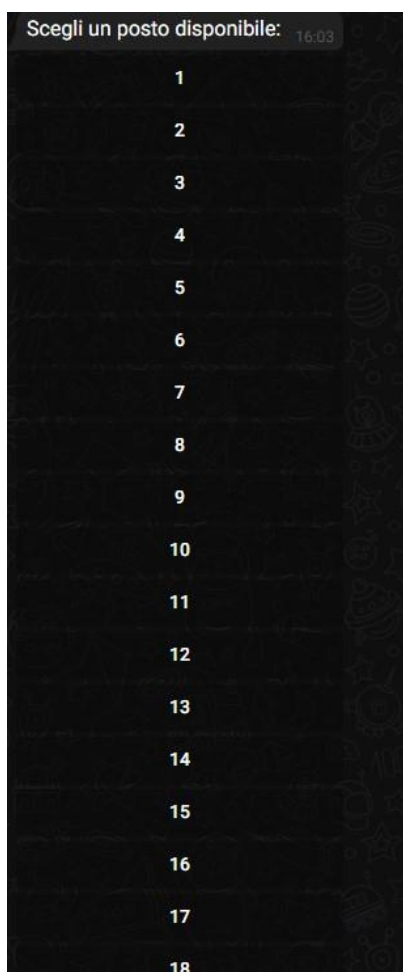
visualizza tutti i corsi: una volta premuto il pulsante si vedrà questa schermata che permette la scelta dell'anno di cui vedere i corsi



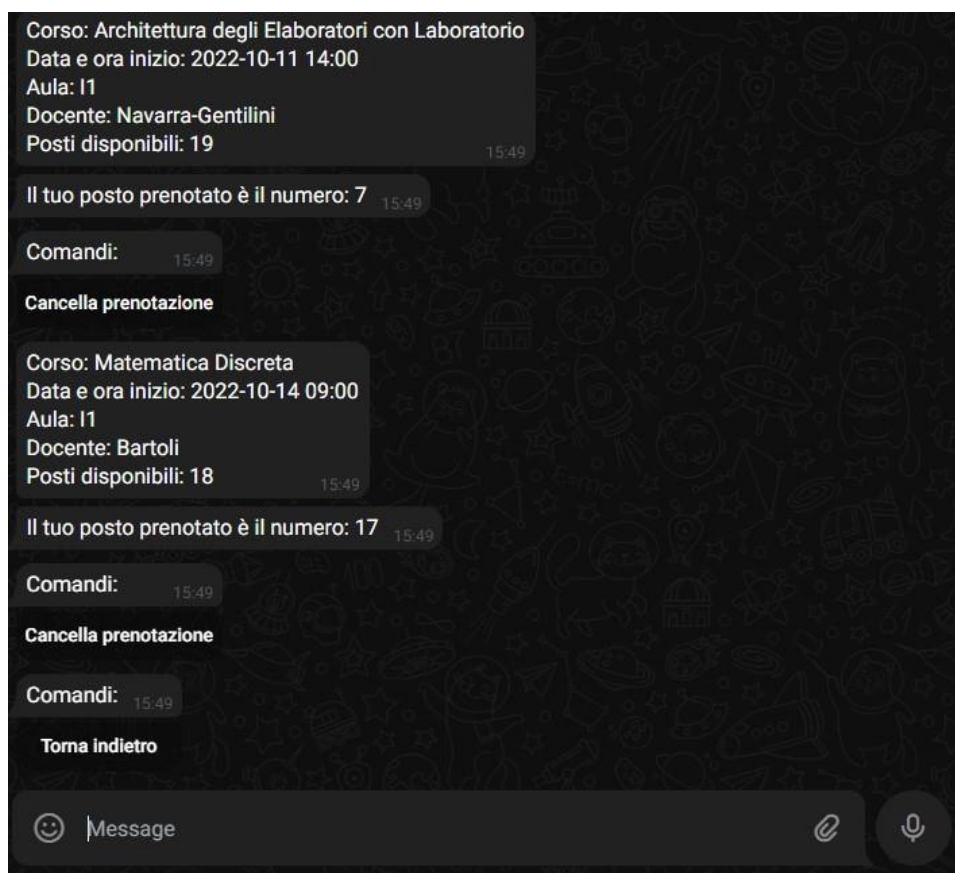
corsi del primo anno: una volta premuto il pulsante, vengono visualizzati tutti i corsi dell'anno scelto con relativo orario e data



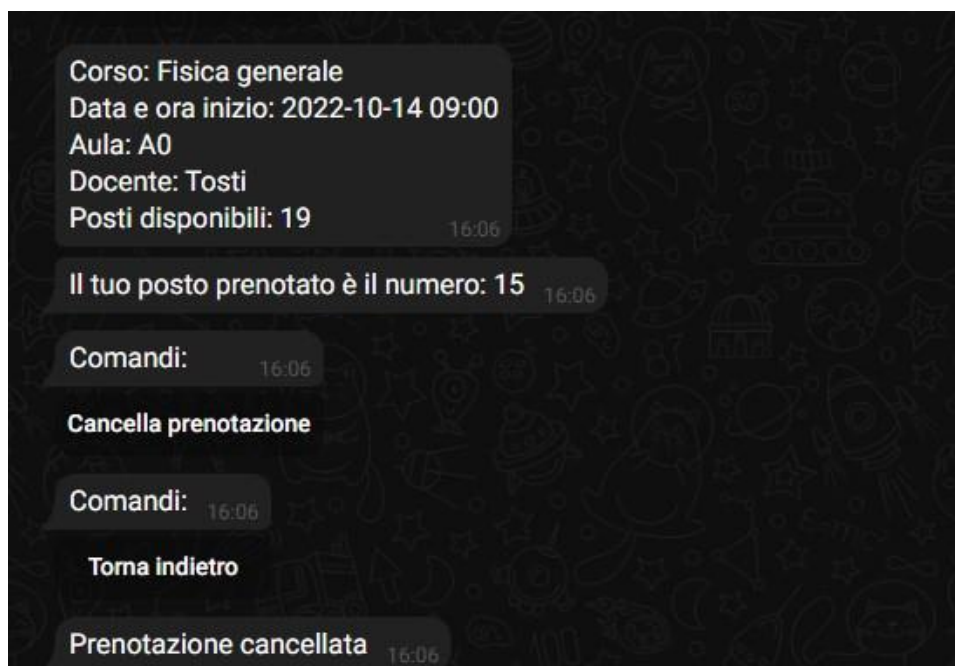
Interfaccia dopo aver cliccato un corso



Interfaccia dopo aver cliccato 'prenotati'.



Esempio di visualizzazione prenotazioni.



Esempio cancellazione prenotazione (possibile solo dopo averne effettuata almeno una)

8 Collegamenti esterni:

[pyTelegramBotAPI](#) -> Documentazione

[sqlite3](#) -> Documentazione

[PrenotazioneLezioneBot](#) -> Link GitHub del progetto