

Documento Requisiti (funzionali e non funzionali)

Babacar Pape Diop, Ahmed Attia Ahmed Ahmed Elabad

Sviluppo di un'applicazione software dedicata alla preparazione degli aspiranti guidatori per l'esame teorico ministeriale della patente di guida (foglio rosa). La piattaforma mira a guidare l'utente attraverso un percorso di apprendimento strutturato e personalizzato, combinando risorse teoriche aggiornate con un'ampia gamma di simulazioni di quiz interattivi, al fine di massimizzare le probabilità di successo all'esame ufficiale.

- Il software sarà suddiviso da una parte teorica di apprendimento, grazie a un manuale e una parte più pratica con dei quiz con il quale sarà possibile esercitarsi
-

SCOPO DEL SERVIZIO

L'applicazione è stata progettata per fornire agli utenti uno strumento completo e veloce per lo studio della teoria della patente. FastLicense offre:

- Accesso a materiale teorico organizzato per argomenti
- Sistema di quiz interattivi per l'esercitazione
- Monitoraggio del progresso di apprendimento
- Notizie e aggiornamenti sul codice della strada
- Supporto e assistenza per gli utenti

DESCRIZIONE

FastLicense è un'applicazione Android dedicata allo studio della teoria per il conseguimento della patente di guida. L'app sarà disponibile 24 ore su 24, 7 giorni su 7, e permetterà agli utenti di:

- Studiare la teoria attraverso contenuti organizzati per argomenti (segnaletica, precedenza, norme di comportamento, ecc.)
- Esercitarsi con quiz simulando l'esame teorico della patente
- Monitorare i progressi con statistiche dettagliate sulle performance
- Ricevere aggiornamenti su modifiche al codice della strada
- Richiedere assistenza per dubbi o problematiche tecniche

REQUISITI

REQUISITI FUNZIONALI

Sistema di Autenticazione

- **Login:** Gli utenti devono poter accedere con username/email e password
- **Logout:** Possibilità di disconnettersi dall'account

Home Page

- Accesso rapido alle sezioni principali

Sezione Teoria

- **Organizzazione per argomenti:** Contenuti divisi per capitoli tematici
 - Segnaletica stradale
 - Norme di precedenza
 - Distanze di sicurezza
 - Limiti di velocità
 - Guida sicura
 - Documenti e assicurazione
 - Elementi del veicolo
 - Comportamento alla guida
- **Contenuti multimediali:** Testi, immagini
- **Ricerca:** Funzione per cercare argomenti specifici

Sezione Quiz

- **Modalità di esercitazione:**
 - Quiz per argomento specifico
 - Quiz simulazione esame
 - Quiz casuale per ripasso generale
- **domande:** Vero/Falso come da esame ministeriale
- **Timer:** Cronometro per simulare le condizioni d'esame
- **Feedback immediato:** Visualizzazione del risultato del test
- **Spiegazioni:** Per ogni domanda, spiegazione della risposta corretta
- **Storico quiz:** Registro dei quiz completati con risultati

Area Utente

- **Profilo personale:** Visualizzazione e modifica dati utente
- **Statistiche dettagliate:**
 - Totale quiz completati
 - Percentuale di successo globale
 - Argomenti più/meno padroneggiati
 - Progressi nel tempo (grafici)
 - Tempo totale di studio
- **Certificati/Badge:** Riconoscimenti per obiettivi raggiunti
- **Impostazioni:** Preferenze notifiche, tema app, privacy

Sezione News

- **Aggiornamenti normativi:** Notizie su modifiche al codice della strada
- **Consigli per l'esame:** Suggerimenti e best practices

- **Articoli tematici:** Approfondimenti sulla sicurezza stradale
- **Notifiche push:** Alert per news importanti

Sezione Assistenza

- **FAQ:** Domande frequenti su app e teoria patente
 - **Contatto supporto:** Form per inviare richieste di assistenza
 - **Tutorial app:** Guide per utilizzare le funzionalità
 - **Feedback:** Sistema per segnalare bug o suggerire miglioramenti
 - **Chat assistenza:** Supporto in tempo reale (se disponibile)
-

REQUISITI NON FUNZIONALI

Usabilità

- Interfaccia intuitiva e user-friendly
- Conformità alle linee guida Material Design per Android
- Accessibilità per utenti con disabilità (dimensione testo, contrasto colori)
- Tempi di risposta inferiori a 2 secondi per ogni azione

Sicurezza

- Crittografia delle password
- Protezione dati personali secondo GDPR
- Sessioni sicure con token di autenticazione
- Backup automatico dei progressi utente

Performance

- Avvio app in meno di 3 secondi
- Caricamento sezioni in meno di 1 secondo
- Ottimizzazione consumo batteria
- Funzionamento offline per contenuti scaricati

Compatibilità

- Android 13.0 (API level 33) o superiore
- Supporto smartphone e tablet
- Orientamento portrait e landscape
- Diverse risoluzioni schermo

Privacy e Consensi

- Richiesta consenso trattamento dati alla registrazione
- Privacy policy e termini d'uso accessibili
- Possibilità di esportare/eliminare i propri dati
- Cookie policy per tracciamento analytics (opzionale)

Contenuti

- Database quiz conforme al Ministero dei Trasporti
- Aggiornamento periodico contenuti teorici
- Sincronizzazione progressi utente

DESCRIZIONI Architettura

"L'app FastLicense è costruita con una struttura 'a tre livelli' MVC, che ci aiuta a tenere ben separate le diverse parti del progetto: il front-end, la logica e il database.

- Frontend (View): È la parte Android in Java che l'utente vede e usa (con l'aiuto di Jetpack components e Material Design). Il suo unico lavoro è presentare le informazioni e interagire con l'utente. Comunica con il server solo tramite le API REST.
- Business Logic (Controller): È il cervello dell'app, sviluppato in Java usando Spring Boot. Qui gestiamo tutto il lavoro 'pesante': elaborare i quiz, calcolare i punteggi, servire la teoria (capitoli e argomenti) e l'autenticazione. Abbiamo configurato il server per essere Stateless (senza sessioni attive), come si vede dalla configurazione di Spring Security.
- DataBase (Model): È dove salviamo tutto (utenti, domande, quiz eseguiti, capitoli). Usiamo un database relazionale come MySQL gestito tramite Spring Data JPA.

Questa separazione è vitale: rende il backend facilmente scalabile e facilita la manutenzione perché ogni componente ha un compito ben definito."

"La sicurezza è gestita a livello del Backend grazie a Spring Security. Le nostre scelte chiave sono:

- Niente Sessioni (Stateless): Abbiamo impostato `SessionCreationPolicy.STATELESS` in `SecurityConfig`. Questo significa che il server non tiene traccia attiva degli utenti tramite sessioni classiche. Idealmente, questo rende il sistema pronto per l'uso di Token JWT (anche se, per ora, l'utente loggato è tenuto in memoria nel `AuthService` per la fase di sviluppo).
- Password Criptate: Le password non vengono mai salvate in chiaro. Appena un utente si registra o fa login, la password viene verificata (o salvata) usando l'algoritmo di crittografia robusto `BCrypt`.
- Accesso Controllato: Solo l'endpoint di login (`/login`) è liberamente accessibile. Tutte le aree che contengono dati sensibili (come `/salvaQuiz` o `/getQuizEseguito/{quizId}`) richiedono l'autenticazione dell'utente. *Nota:* La configurazione Spring attuale permette provvisoriamente l'accesso a tutti (`.anyRequest().permitAll()`), ma per la versione finale dovrà essere ristretta a `anyRequest().authenticated()`."

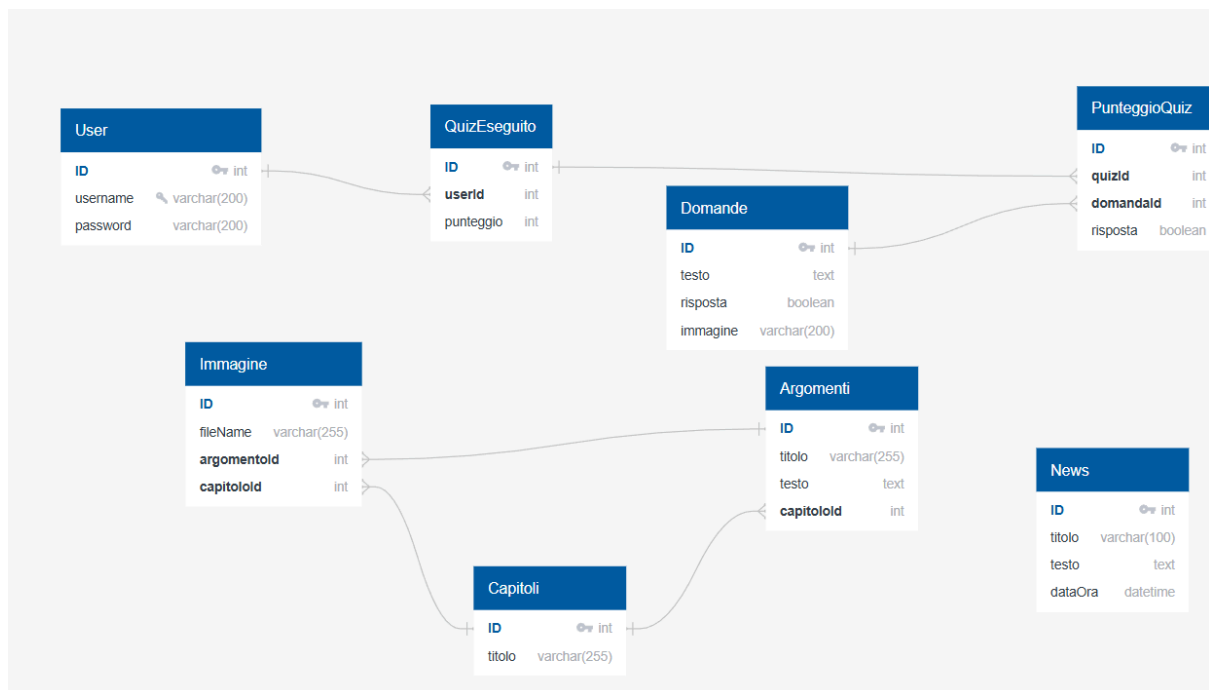
L'utente esegue un quiz da 30 domande casuali (ricevute tramite `/getDomande`). La parte più importante è il salvataggio:

- Salvataggio: Il Client Android invia le risposte date al Backend (/salvaQuiz).
- Correzione: Il Backend (Spring Boot) si occupa di controllare ogni singola risposta. Confronta la risposta fornita dall'utente con la risposta corretta che ha nel database.
- Punteggio: Il sistema calcola il punteggio finale (risposte corrette su 30) e lo memorizza sia come risultato totale del quiz (QuizEseguito) sia per ogni singola domanda (PunteggioQuiz).

Utilizziamo il pattern DTO (Data Transfer Object) in modo massiccio perché è una buona pratica di programmazione. Serve a due scopi principali:

1. Sicurezza e Pulizia: Inviando all'app mobile solo i dati strettamente necessari per una data operazione. In questo modo, evitiamo di esporre dati interni inutili o sensibili (come l'intera entità del database).
2. Disaccoppiamento: Il Backend può cambiare la sua struttura interna (le entità) senza dover necessariamente rompere il Frontend (l'app Android).

DATABASE



WIREFRAME E DESCRIZIONI ACTIVITY

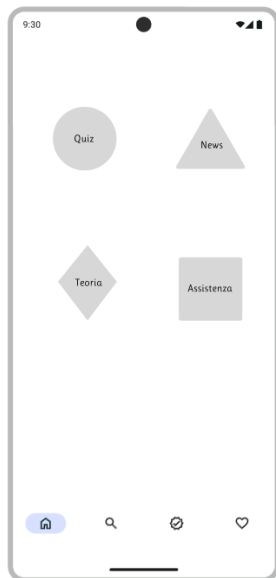
Pagina di login

Questa è la schermata iniziale per accedere all'app. Si presenta con due campi per inserire l'Username/Email e la Password, e un pulsante "Invia" per effettuare il login. È il punto di accesso per tutti gli utenti.



home:

La Home Page è il punto di partenza dell'esperienza utente. Permette un accesso rapido e visivo alle quattro sezioni principali dell'app: Quiz, News, Teoria e Assistenza, rappresentate con icone e forme chiare. In basso, è presente una barra di navigazione fissa che agevola il passaggio tra le funzionalità principali.

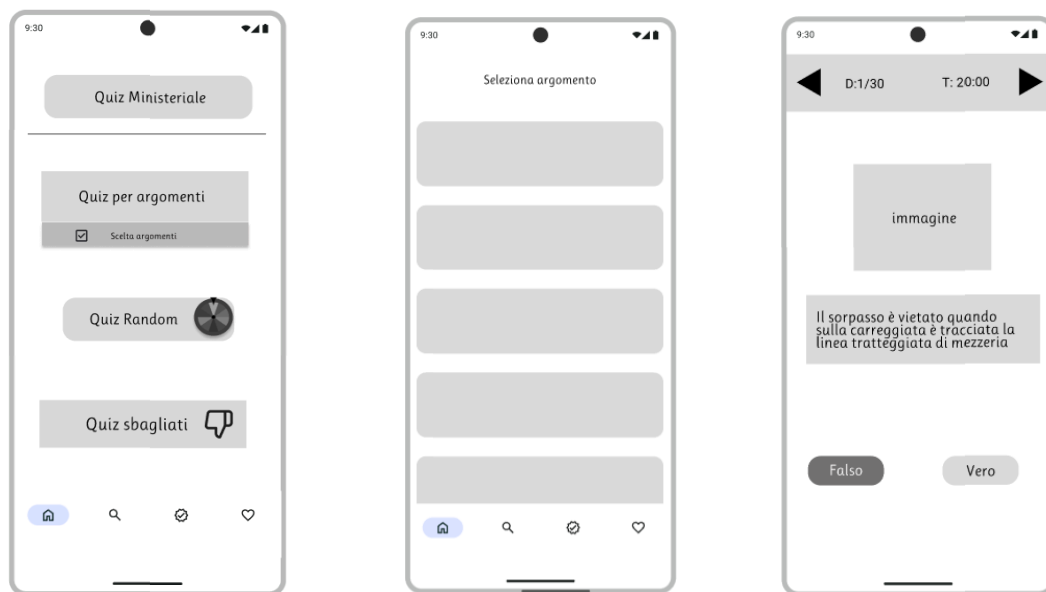


quiz:

Menu Quiz: Offre le diverse modalità di esercizio: Quiz Ministeriale (simulazione d'esame ufficiale), Quiz per argomenti (per un futuro aggiornamento), Quiz Random (per un ripasso generale), la possibilità di fare quiz mirati sugli Errori commessi in precedenza (IN un futuro aggiornamento).

Selezione Argomento: Permette all'utente di scegliere i capitoli o i temi specifici sui quali desidera concentrare la sessione di esercitazione, se ha selezionato la modalità .

Esecuzione Quiz: Questa schermata mostra la domanda in corso. In alto sono visibili il **progresso** (es. Domanda 1 di 30) e il **Timer** (es. 20:00). Al centro c'è lo spazio per l'eventuale **immagine** e subito sotto il testo della domanda. L'utente risponde selezionando **"Falso"** o **"Vero"**.

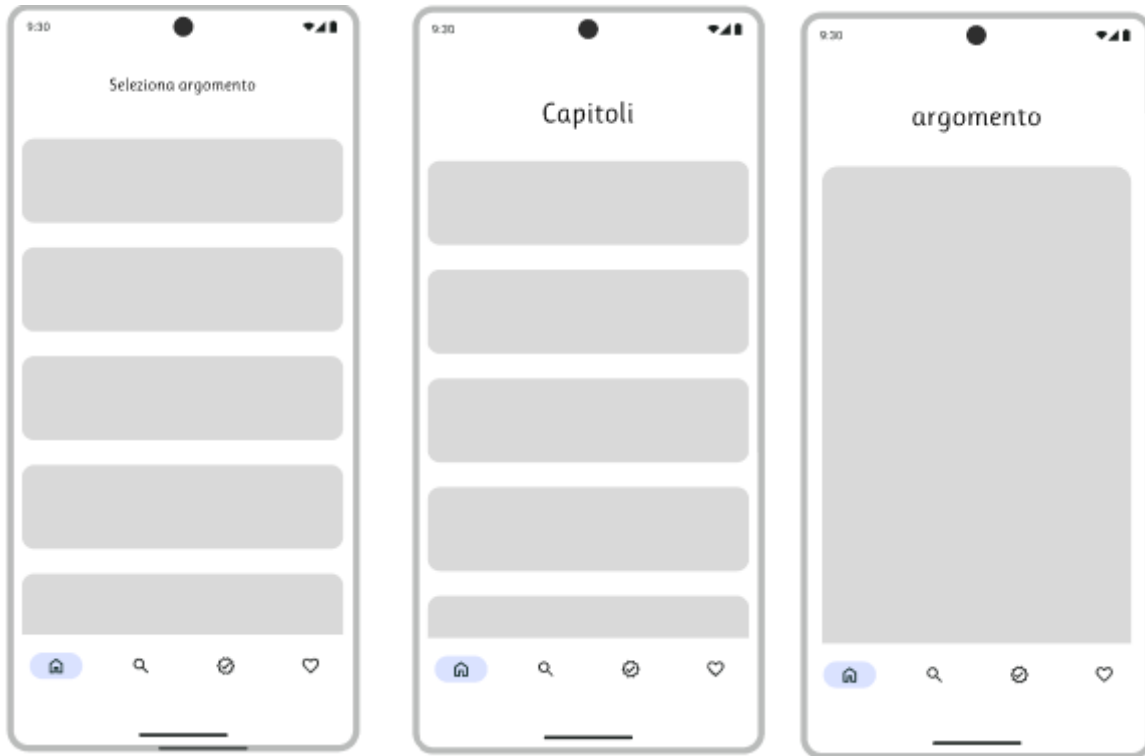


Teoria:

La Teoria è strutturata in un percorso di apprendimento organizzato:

- Capitoli: La sezione si apre sulla lista principale, mostrando i Capitoli tematici della teoria della patente.

- Lista Argomenti: Dopo aver selezionato un Capitolo, questa schermata mostra la lista dettagliata degli Argomenti contenuti in quel capitolo specifico.
- Dettaglio Argomento: Schermata che visualizza il testo completo dell'argomento scelto e include lo spazio per mostrare immagini, schemi o contenuti multimediali di supporto.



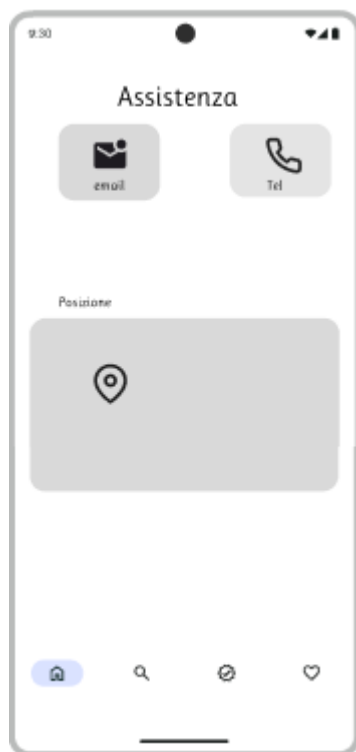
Area utente:

L'Area Utente funge da profilo personale. Qui l'utente può vedere il proprio Nome utente ed email e, cosa fondamentale, in futuro si potrà accedere alle proprie Statistiche dettagliate (come Quiz Svolti, Media Errori e i Progressi nel tempo). Un pulsante centrale permette di modificare i dati personali.



Assistenza:

Questa schermata è dedicata al supporto. Fornisce i contatti rapidi per ricevere assistenza, con due pulsanti dedicati per l'invio di una Email e per avviare una Telefonata. Inoltre, include una sezione "Posizione" che mostra la location (probabilmente una mappa o un indirizzo di riferimento) per il supporto fisico.



| METODO | ENDPOINT | PARAMETRI | RISPOSTA di Successo (200 OK) | NOTE |
|--------|---------------------------|--|---|---|
| POST | /login | Body (JSON): username, password | {"risultato": "Login effettuato con successo!"} | Effettua l'accesso utente. Restituisce 401 Unauthorized in caso di credenziali non valide. |
| GET | /news | / | List<NewsDto> | Restituisce tutte le notizie, ordinate dalla più recente. |
| GET | /getDomande | / | { "domande": List<DomandeDto> } | Restituisce 30 domande casuali per l'esecuzione di un quiz. Non include la risposta corretta. |
| POST | /salvaQuiz | Body (JSON): List<QuizEseguitoDto> (Contiene risposte date per ogni domanda) | { "quizSalvati": List<{quizId, punteggio}> } | Salva i risultati del quiz per l'utente loggato e calcola il punteggio. Richiede autenticazione (401 se non loggato). |
| GET | /getQuizEseguito/{quizId} | Path: quizId (Integer) | Dettagli del quiz, punteggi e risposte date/corrette. | Restituisce il dettaglio di un quiz eseguito specifico, inclusa la correzione. Richiede autenticazione. Restituisce 404 se non trovato o non |

| | | | | |
|-----|---|---|---|---|
| | | | | appartenente all'utente. |
| GET | /getCapitoli | / | { "capitoli": List<CapitoliDto> } | Restituisce la lista di tutti i capitoli con titolo e (opzionale) l'immagine principale. |
| GET | /getArgomento/{capitolold} | Path: capitolold (Integer) | { "argomenti": List<ArgomentiDto> } | Restituisce la lista di tutti gli argomenti appartenenti a quel capitolo. Restituisce 404 se il capitolo non è trovato. |
| GET | /getArgomentoDettaglio/{capitolold}/{argomentold} | Path: capitolold (Integer), argomentold (Integer) | { "argomento": ArgomentiDto } | Restituisce il testo completo e la lista dei nomi dei file immagine di un argomento specifico. Restituisce 404 se l'argomento/capitolo non è trovato. |

Conclusione

Il Documento di Requisiti ha definito FastLicense come una piattaforma di apprendimento completa per la patente di guida.

- Obiettivo: Fornire un percorso strutturato che combina Teoria (organizzata per argomenti), Esercitazioni (simulazioni e quiz mirati) e Monitoraggio dei Progressi, massimizzando le probabilità di successo all'esame.
- Architettura e Sicurezza: L'architettura MVC a tre livelli (Spring Boot/Android) assicura scalabilità e manutenibilità. La sicurezza è garantita dalla crittografia delle password (BCrypt) e da un'autenticazione Stateless, garantendo affidabilità e conformità al GDPR.

