

Linee Guida di Sviluppo – Progetto App Padel

Questo documento raccoglie le linee guida per lo sviluppo del progetto finale di Ingegneria del Software: un'applicazione per organizzare partite di padel tra sconosciuti. Il documento è destinato a guidare lo sviluppo iniziale su piattaforme come Replit, fornendo una base minima funzionante con interfaccia grafica semplice e funzionalità essenziali.

Obiettivi principali

- Realizzare un'app con architettura client/server basata su MVC.
- Prevedere almeno due design pattern documentati (Observer, Strategy, Singleton).
- Garantire una UI semplice che permetta di usare subito l'app anche in forma prototipale.
- Implementare solo il necessario all'inizio (autenticazione e funzioni complesse come placeholder).
- Raggiungere rapidamente una versione dimostrabile e iterare successivamente.

Requisiti funzionali

- 1 Registrazione utente con livello dichiarato (placeholder iniziale, login semplificato).
- 2 Creazione e gestione di partite di due tipi: fisse (programmate) e proposte (dagli utenti).
- 3 Iscrizione e cancellazione da partite, con limite massimo di 4 giocatori.
- 4 Conferma automatica di una partita al raggiungimento di 4 iscritti.
- 5 Gestione stato partite: WAITING, CONFIRMED, FINISHED.
- 6 Feedback post-partita con valutazione livello e commento.
- 7 Aggiornamento livello percepito di un utente in base ai feedback.
- 8 Contatore partite giocate aggiornato quando una partita passa a FINISHED.
- 9 Ricerca e filtro partite per livello richiesto.
- 10 Ordinamento partite con Strategy (per data, popolarità, livello).
- 11 Notifiche con Observer + Singleton (es. partita confermata, partita conclusa).

Requisiti non funzionali

- Architettura MVC e programmazione a oggetti (Java, Spring Boot, oppure stack equivalente).
- UI semplice (Thymeleaf o HTML minimale) con navigazione base.
- Password e login come placeholder iniziali, con implementazione reale in fasi successive.
- Database relazionale (H2 per sviluppo).
- Test unitari (JUnit) con coverage $\geq 80\%$.
- Documentazione UML: Use Case, Class Diagram, almeno 2 Sequence Diagram.
- Relazione finale ≤ 40 pagine (per progetto individuale).

Piano di sviluppo iterativo

- Fase 0 – Setup progetto e repo.
- Fase 1 – Modellazione dati (User, Match, Registration, Feedback).
- Fase 2 – UI semplice: login placeholder, lista partite, profilo utente.
- Fase 3 – CRUD partite e iscrizioni (limite 4).
- Fase 4 – Pattern Observer per notifiche conferma/chiusura.
- Fase 5 – Feedback e calcolo livello percepito.
- Fase 6 – Strategy per ordinamento partite.
- Fase 7 – Singleton NotificationService.
- Fase 8 – Test unitari + coverage $\geq 80\%$.

- Fase 9 – Documentazione e UML.
- Fase 10 – Versione dimostrabile + demo.

Queste linee guida forniscono una roadmap chiara e pragmatica: partire da una UI funzionante, con funzioni core semplificate, e iterare aggiungendo autenticazione completa e altre funzionalità avanzate solo in seguito. L'obiettivo è garantire rapidamente una versione dimostrabile e usabile, in linea con i requisiti accademici del progetto.