

Ecco un esempio di testo d'esame completo per un corso biennale post-diploma da Web Developer, centrato sul tema "Prenotazione biglietti aerei".

📌 ESAME FINALE – CORSO BIENNALE WEB DEVELOPER Tema: Sistema di prenotazione biglietti aerei Durata prevista: 6–8 ore Strumenti consentiti: IDE (es. VS Code), browser, database locale, documentazione ufficiale, Git Tecnologie richieste: HTML, CSS, JavaScript, PHP/Java/Python, SQL, MySQL/PostgreSQL, framework a scelta (Laravel, Spring Boot, Django...)

📄 TRACCIA Realizza una web application funzionante per la gestione della prenotazione di biglietti aerei, che consenta:

La visualizzazione dei voli disponibili

La prenotazione da parte dell'utente

L'amministrazione dei voli e dei biglietti da parte di un operatore

La gestione dell'autenticazione e autorizzazione

🔑 FUNZIONALITÀ MINIME RICHIESTE 👤 1. Anagrafica voli Ogni volo deve contenere:

Numero volo

Aeroporto di partenza e arrivo

Data e ora

Posti disponibili

Prezzo del biglietto

📄 2. Prenotazione biglietti Gli utenti registrati possono:

Visualizzare la lista dei voli

Filtrare per data o destinazione

Prenotare uno o più biglietti

Visualizzare le prenotazioni effettuate

🔑 3. Autenticazione Login e registrazione utenti

Area riservata:

Utente: prenotazioni personali

Amministratore: gestione voli, gestione utenti, report prenotazioni

📄 4. CRUD amministrativo (backend o interfaccia web) Inserimento, modifica, eliminazione dei voli

Report settimanale dei voli più prenotati

📄 STRUTTURA DATI MINIMA (schema semplificato) utenti(id, nome, email, password, ruolo)

voli(id, numero, aeroporto\_partenza, aeroporto\_arrivo, data\_ora, posti\_disponibili, prezzo)

prenotazioni(id, id\_utente, id\_volo, data\_prenotazione, n\_posti)

💡 SUGGERIMENTI TECNICI Utilizza un pattern MVC (o simile)

Proteggi le rotte amministrative con middleware o controlli lato server

Puoi usare AJAX per migliorare l'esperienza utente

Inserisci validazioni sia lato client che lato server

Bonus: generazione PDF della prenotazione / invio email

☑ CRITERI DI VALUTAZIONE Criterio Peso Funzionalità implementate 40% Qualità del codice 20% Architettura e organizzazione 15% UI/UX e responsive design 10% Uso corretto di Git 5% Documentazione del progetto 10% 📦 CONSEGNA Codice sorgente via GitHub/git zip

File README.md con:

Descrizione progetto

Tecnologie utilizzate

Istruzioni per avvio locale

Eventuale dump del database (es. dump.sql)