**Fac-Simile Prova d’esame Intermedia - BIT 2020**

**Introduzione**

La Casa Editrice Mondadori sta pensando di sviluppare una applicazione web per permettere ai suoi utenti di avere la propria libreria in digitale, andando a segnare i libri letti, e quelli che si vorrebbero leggere, dando la possibilità di recensire i libri che si hanno letto e di creare liste personalizzate di libri, in stile Amazon.

Dato che il loro segmento di mercato non è particolarmente avvezzo con la tecnologia mobile, hanno pensato di lanciare inizialmente il prodotto come una piattaforma web.

Lo sviluppo consta di due parti separate:

* Analisi Problema & UI/UX
* Design Backend & Sviluppo Prototipo

**Desiderata del prototipo:**

* Possibilità per il superadmin di creare utenti di tipo Casa Editrice e Lettore
* Possibilità per la Casa Editrice di inserire un Libro (solo dati, no immagine copertina)
* Possibilità per un utente Lettore di inserire un Libro nella lista dei libri letti
* Possibilità per un utente Lettore di inserire un Libro nella lista dei libri da leggere
* Possibilità per un utente Lettore di visualizzare la propria liste “Da Leggere”
* Possibilità per un utente Lettore di visualizzare la propria lista “Letti”
* **Extra:** Possibilità per un utente Lettore di creare Liste personalizzate
* **Extra:** Possibilità per un utente Lettore di aggiungere Libri alle liste Personalizzate
* **Extra:** Possibilità per un utente Lettore di visualizzare il contenuto delle proprie liste Personalizzate

**Consegne**

*Parte “Analisi Problema & UI/UX”:*

1. Definire la lista delle funzionalità che avrà l’applicazione, sulla base delle richieste presenti nell’introduzione e nei *desiderata del prototipo*. Questa lista deve essere presentata in forma di elenco puntato. E’ possibile aggiungere anche una descrizione *discorsiva*, se può aiutare in qualche modo, ma verrà valutata solamente la lista. (3 punti)
2. Creare un wireframe, abbozzando le schermate necessarie a risolvere tutte le funzionalità descritte nel punto precedente. Il wireframe, per sua natura, non ha bisogno di essere dettagliato a livello grafico, deve semplicemente mostrare come e dove si intendono posizionare i vari blocchi (testi, pulsanti, immagini, ecc) e come questi vadano a *risolvere* il problema che si sta analizzando. (4 punti)
3. A partire dal wireframe, disegnare il mockup *pixel perfect,* andando a definire uno stile consistente per le varie schermate e popolando i vari testi e/o contenuti utilizzando il Lorem Ipsum (3 punti)
4. **Extra:** creare logo, brand identity e strategy, accompagnando le grafiche create con una breve descrizione delle motivazioni alla base delle scelte (5 punti)

*Parte “Design Backend & Sviluppo Prototipo”:*

1. Definire la struttura del database del progetto, tenendo conto dei requisiti minimi descritti al punto 1 della prima parte e delle *desiderata del prototipo* (4 punti)
2. Descrivere in maniera dettagliata le azioni CRUD che possono effettuare i vari tipi di utenti, descrivendole in forma di elenco. (3 punti)
3. Creare la struttura database usando le migrazioni di Laravel, definendo le tabelle e le relazioni tra tabelle necessarie. (3 punti)
4. **Extra:** creare un prototipo usando Laravel & Vue che permetta di soddisfare le *desiderata del prototipo*. Consiglio: create un’autenticazione semplice, senza controllo dei cookie, usando il LocalStorage per tenere le informazioni circa l’utente connesso. (5 punti)

**Modalità d’esame**

* L’esame può essere svolto nella maniera che si preferisce, non c’è un ordine preciso nelle domande. La progressione delle richieste è stata pensata per guidare la soluzione, ma è possibile saltare alcuni punti o completarli in seguito.
* Al fine di ricevere i punti della risposta è necessario che sia chiaramente riportato il numero della domanda e il nome della sezione, ad esempio “Parte ‘Design BackEnd & Sviluppo Prodotto’ - Domanda 2”. In caso contrario, non verranno assegnati i punti.
* Non ci saranno penalità se non si completano alcune parti.
* Il punteggio massimo raggiungibile è di 30 punti.
* La prova verrà considerata sufficiente con 8 punti.
* I punti extra vengono calcolati anche in assenza delle risposte precedenti. A titolo di esempio è possibile essere più che sufficienti svolgendo solamente i due punti “Extra”.
* E’ fortemente consigliato (leggi obbligatorio) utilizzare il template presente nella pagina seguente per lo svolgimento della prova.

**Prova d’esame Intermedia - BIT 2020**

**Nome Cognome**

**Risoluzione**

*Parte “Analisi Problema & UI/UX”:*

1. Il prototipo avrà le seguenti funzionalità:
   1. L’applicazione avrà una parte di login divisa per lettore o editore
   2. X LETTORE: Una schermata centrale con delle liste, a forma di libreria, divise in
      1. Libri letti
      2. Libri da leggere
      3. Libri preferiti
      4. – altre liste personalizzate –
   3. X LETTORE: Cliccando su un libro a destra comparirà un riquadro scrollabile con tutte le info del libro, le recensioni e la possibilità di fare una recensione o aggiungere il libro a una lista (pulsanti “letto”, “da leggere”, “aggiungi ad altra lista”)
   4. X LETTORE: Oltre alla sezione libreria ci sarà una sezione dove cercare i libri, cliccarli e vederne i dettagli sempre col riquadro a destra
   5. X CASAEDITRICE: Oltre al Login avrà una semplice schermata per aggiungere nuovi libri e modificare info libri già caricati
   6. X SUPERADMIN: Ci sarò log in personalizzato e una sezione dove può creare nuove utenze con credenziali da poi consegnare alle case editrici richiedenti
2. Vedi allegato “wireframe”
3. Vedi cartella allegata “mockup"
4. Risposta 4

*Parte “Design Backend & Sviluppo Prototipo”:*

1. Il database sarà composto da 5 tabelle:
   1. USERS (ID, email,pass, name, role\_id) – role\_id FK 🡪roles.id
   2. ROLES (ID, role) – i ruoli saranno predefiniti “lettore”, “editore”, “superadmin”
   3. BOOKS (ID, title, desc, author, year, editor\_id) – editor\_id FK 🡪 users.id
   4. LISTS (ID, name, lettore\_id) – lettore \_id FK 🡪users.id
   5. BOOK-LIST-USER (ID, lettore\_id, book\_id, list\_id)
      1. lettore\_id FK 🡪 users.id
      2. book\_id FK 🡪 books.id
      3. list\_id FK 🡪 lists.id
2. CRUD LETTORE:
   1. Creare nuove liste
   2. Aggiungere/rimuovere libri alle liste
   3. Aggiungere/rimuovere recensioni fatte dallo stesso

CRUD EDITORE:

* 1. Aggiungere nuovi libri
  2. Modificare info libri già caricati

CRUD SUPERADMIN

* 1. Aggiungere nuove utenze lettore o editore

1. Vedi allegato codice 1
2. Risposta 4

**Allegati**

1. Allegato di codice 1: massimo-piazza-esame/code/server/database/migrations
2. Allegato di codice 2: /path/del/codice
3. Allegato immagine 1: /massimo-piazza-esame/mockup
4. Allegato immagine 2: /massimo-piazza-esame/wireframe.png

*(NDR: Per le immagini è possibile inserirle direttamente nel documento. Il codice può essere inserito sotto forma di immagine nel documento - screenshot dell’editor -* ***ma deve essere allegato al presente documento****, inserendolo nello ZIP che si consegnerà a fine prova)*