

## Tema n.1: Web application per la gestione dei tirocini degli studenti (traccia estratta)

L'Ateneo intende sviluppare un'applicazione web responsive per la gestione dei tirocini dei corsi di studio dell'area medica.

Gli studenti dei corsi dell'area medica devono partecipare ad un elenco predefinito di tirocini presso strutture sanitarie esterne convenzionate con l'Università. I turni dei tirocini hanno durata variabile, a seconda della materia, da 1 a 3 settimane e si svolgono nel corso dell'anno accademico su un arco di 40 settimane da settembre fino a giugno.

Ogni struttura ospitante offre disponibilità per ospitare, ogni turno, un numero massimo di studenti.

L'applicazione web deve permettere:

- Alle segreterie studenti:
  - Definire il calendario dei tirocini, specificando per ogni materia:
    - Durata dei turni (numero di settimane);
    - Numero di posti disponibili per ogni struttura ospitante; il numero di posti disponibili può variare ad ogni turno
  - Assegnare gli studenti ai tirocini, tenendo conto delle loro preferenze e delle disponibilità delle strutture ospitanti.
- Agli studenti:
  - Consultare l'elenco dei tirocini assegnati, con le informazioni relative a:
    - Struttura ospitante
    - Periodo di svolgimento
    - Tutor aziendale
    - Attività previste
  - Comunicare con le segreterie studenti e le strutture ospitanti.
- Alle strutture ospitanti:
  - Conoscere l'elenco degli studenti assegnati di settimana in settimana.
  - Gestire la presenza degli studenti.
  - Fornire feedback agli studenti.
  - raccogliere le presenze dei tirocinanti durante il periodo di tirocinio.
  - valutare i tirocinanti alla fine del periodo di tirocinio.

Il candidato:

1. discuta l'architettura dell'applicazione, in termini di:
  - a. stack applicativi (tecnologie, linguaggi, framework applicativi), descrivendo anche più scenari ed eventuali vantaggi/svantaggi;
  - b. architettura generale (livelli applicativi);
2. descriva la struttura del database dell'applicazione in termini di
  - a. elenco delle tabelle e dei campi;
  - b. chiavi primarie;
  - c. relazioni;
3. sulla base della struttura del database ipotizzata scriva le seguenti query SQL:
  - a. Numero di studenti per materia e struttura ospitante in una determinata settimana;
  - b. Turni di una determinata materia che non hanno studenti assegnati
4. proponga eventuali miglioramenti apportabili alle funzionalità;

## Tema n.2: Web application per la gestione degli appelli d'esame

L'Ateneo intende sviluppare un'applicazione web responsive per la gestione degli appelli d'esame.

Ogni anno accademico viene nominata per ogni materia d'esame una commissione composta dal presidente e dai membri (tutti docenti del corso). Ad ogni sessione d'esame il presidente stabilisce le date degli appelli; gli studenti frequentanti possono iscriversi agli appelli durante il periodo di iscrizione della durata di 30 giorni, fino a 5 giorni prima dell'appello.

L'applicazione web deve permettere:

- alle segreterie studenti:
  - di stabilire la composizione delle commissioni di ogni materia d'esame;
  - di stabilire le date delle sessioni d'esame e il numero minimo di appelli che ogni docente deve inserire;
  - di consultare gli appelli inseriti dai docenti e le liste degli iscritti
  - di modificare la composizione della commissione per una singola data di appello nel caso in cui uno dei membri non possa partecipare;
  - di assegnare l'aula in cui si svolgerà l'appello;
- ai docenti:
  - di aggiungere le date degli appelli; ogni appello può essere strutturato in più turni: ogni turno è definito da una data, un orario di convocazione, una tipologia (orale, scritto, CBT - computer based testing, ...) e le note per gli studenti;
  - di consultare la lista degli iscritti;
  - di inserire le valutazioni degli studenti al termine dell'esame, dando eventualmente la possibilità di accettare o rifiutare il voto ed un termine entro cui lo studente si può pronunciare;
- agli studenti:
  - di consultare la lista degli appelli per ogni sessione d'esame;
  - di iscriversi all'appello, inserendo eventuali note per il docente;
  - di rinunciare all'iscrizione entro i termini consentiti (il sistema deve tenerne traccia);
  - di consultare le valutazioni inserite dai docenti ed eventualmente accettare o rifiutare il voto;

Il candidato:

1. discuta l'architettura dell'applicazione, in termini di:
  - a. stack applicativi (tecnologie, linguaggi, framework applicativi), descrivendo anche più scenari ed eventuali vantaggi/svantaggi;
  - b. architettura generale (livelli applicativi);
2. descriva la struttura del database dell'applicazione in termini di
  - a. elenco delle tabelle e dei campi;
  - b. chiavi primarie;
  - c. relazioni;
3. sulla base della struttura del database ipotizzata scriva le seguenti query SQL:

- a. numero di studenti iscritti non rinunciatari ad una data di appello definita, suddivisi per prova;
  - b. materie d'esame per cui non è stato definito il numero minimo di 3 date d'appello per sessione;
4. proponga eventuali miglioramenti apportabili alle funzionalità;

## Tema d'esame n. 3: Web application per l'iscrizione ai test di ammissione

L'Ateneo intende sviluppare un'applicazione web responsive per la gestione dei test di ammissione dei corsi a numero chiuso.

Ogni anno accademico l'Università bandisce i concorsi di ammissione ai propri corsi di studio a numero chiuso.

Ogni bando è valido per uno o più corsi di studio.

In caso di bandi per più corsi di studio i candidati devono specificare un massimo di 3 preferenze; dopo la pubblicazione della graduatoria finale i candidati verranno chiamati a turni, chiedendo di confermare l'immatricolazione ad uno dei corsi che hanno scelto, fino ad esaurimento dei posti disponibili.

L'applicazione web deve permettere:

- alle segreterie studenti:
  - di stabilire l'elenco dei bandi, definendo:
    - le date di svolgimento delle prove di ammissione (almeno una);
    - il periodo di iscrizione;
    - l'elenco dei corsi di studio a cui da titolo e il numero di posti disponibili per ogni corso (corsi a numero chiuso);
    - la commissione che esaminerà le prove (1 presidente, 1 segretario e più membri);
  - di consultare l'elenco degli iscritti ai bandi verificando il pagamento della quota di iscrizione;
  - di pubblicare la graduatoria generale, calcolata usando la media dei voti delle prove;
- ai candidati:
  - di iscriversi ai bandi, specificando:
    - le eventuali preferenze quando il bando riguarda più corsi di studio;
    - di dichiarare eventuali disabilità che richiedono ausili (calcolatrice, schermo ad alto contrasto, etc)
  - pagare la quota di iscrizione
  - rinunciare all'iscrizione (la rinuncia dovrà essere tracciata)
  - di consultare la propria posizione in graduatoria dopo la pubblicazione;
  - di scegliere il corso di immatricolazione nel caso siano in posizioni di graduatoria utile;
- alle commissioni
  - di consultare l'elenco degli iscritti per i bandi di propria competenza e verificare la presenza di candidati che hanno bisogno di ausili;
  - di inserire gli esiti del test di ammissione; ogni studente avrà un voto per ogni prova definite nel bando;

Il candidato:

1. discuta l'architettura dell'applicazione, in termini di:
  - a. stack applicativi (tecnologie, linguaggi, framework applicativi), descrivendo anche più scenari ed eventuali vantaggi/svantaggi;
  - b. architettura generale (livelli applicativi);

2. descriva la struttura del database dell'applicazione in termini di:
  - a. elenco delle tabelle e dei campi;
  - b. chiavi primarie;
  - c. relazioni;
3. sulla base della struttura del database ipotizzata scriva le seguenti query SQL:
  - a. numero di iscritti non rinunciatari per ogni bando e corso di preferenza;
  - b. il voto minimo, massimo e medio per ogni prova degli studenti che si sono immatricolati ad un determinato corso.
4. proponga eventuali miglioramenti apportabili alle funzionalità;