

Requisiti. L'applicazione da progettare riguarda una parte del sistema di gestione di elezioni in un collegio elettorale del comune di Noncè. Alle elezioni vengono presentate delle liste, ciascuna con un nome ed un simbolo (una stringa che identifica il file con l'immagine). Le liste sono suddivise in liste per l'elezione del sindaco e liste per l'elezione del consiglio comunale. Ogni lista per il sindaco contiene un solo candidato. Mentre ogni lista per il consiglio contiene un insieme non vuoto e ordinato di candidati. Una lista per il consiglio può essere collegata ad una (e non più di una) lista per il sindaco. Ogni candidato è caratterizzato dal nome e dall'indirizzo web del suo blog (una stringa). Un candidato può presentarsi in al più una lista per il consiglio ed al più una lista per il sindaco. Nel comune di Noncè ci sono diversi seggi elettorali ciascuno contraddistinto da un codice e dal numero di cittadini aventi diritto al voto iscritti al seggio. Data una lista ed un seggio è di interesse memorizzare quanti voti ha preso la lista in quel seggio, ma solo se il numero di voti è maggiore di 0.

L'ufficio elezioni è interessato ad effettuare diversi controlli sui seggi e le liste, in particolare:

- data una lista ℓ , restituire l'insieme dei candidati a sindaco contenuti in ℓ , cioè se ℓ è una lista per il sindaco restituire l'insieme costituito dal solo candidato presente in essa, se ℓ è una lista per il consiglio restituire l'insieme formato dai candidati presenti in essa che sono anche candidati in una lista per il sindaco;
- dato un seggio s , restituire la percentuale di votanti, cioè il rapporto tra voti ottenuti dalle varie liste nel seggio s e numero di iscritti ad s .

Domanda 1. Basandosi sui requisiti riportati sopra, svolgere la fase di analisi producendo lo schema concettuale in UML per l'applicazione e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

Domanda 2. Svolgere la fase di progetto, illustrando i prodotti rilevanti di tale fase e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

È obbligatorio solo progettare gli algoritmi e definire le responsabilità sulle associazioni.

Domanda 3. Svolgere la fase di realizzazione, producendo un programma JAVA e motivando, qualora ce ne fosse bisogno, le scelte effettuate.

È obbligatorio realizzare in JAVA solo i seguenti aspetti dello schema concettuale:

- le classe `ListaConsiglio` e `Candidato`, le eventuali classi da cui sono derivate, e tutte le associazioni che le legano;
 - il primo use case.
-