

Requisiti

Si realizzi un sistema software per la gestione di un magazzino automatizzato di articoli da ufficio. Il sistema dovrà consentire ai dipendenti di una multinazionale dell'informatica di prenotare dal proprio ufficio articoli appartenenti a differenti categorie di prodotti (cancelleria, toner, chiavette usb, ecc.) e quindi di riceverli comodamente dal magazziniere più vicino. I prodotti hanno un formato standard (nome, categoria, descrizione, prezzo, disponibilità a magazzino, numero massimo di unità ordinabili, ecc.), e sono fabbricati da differenti produttori (di cui il sistema deve tenere traccia nel dettaglio per eventuali reclami) e riforniti da diversi fornitori (di cui il sistema deve tenere traccia per effettuare gli ordini di rifornimento dei magazzini quando le scorte scarseggiano). Un dipendente della multinazionale si connette al sistema con le proprie credenziali e consulta i prodotti disponibili con le relative quantità di articoli a magazzino e costi (questi si riferiscono a quelli disponibili nel magazzino a lui più vicino, ma potrebbe anche scegliere magazzini più distanti, con un costo di consegna quindi più elevato). Quando decide di equipaggiarsi di nuovi articoli, il dipendente si connette al sistema con le proprie credenziali, sceglie dal catalogo i prodotti di suo interesse e le relative quantità. Quando ha completato l'elenco, lo conferma indicando il progetto (la multinazionale ha un elenco esplicito di progetti che tiene traccia del nome del progetto e del nome del capo progetto) sul quale appoggiare la spesa (la merce prenotata non risulterà quindi più disponibile agli altri dipendenti); quindi, stampa una distinta delle merce da ricevere che dovrà consegnare al magazziniere quando questi si recerà nel suo ufficio per la consegna. Un dipendente può cambiare idea, modificando l'elenco prima di confermarlo definitivamente. Il magazziniere, più volte al giorno, accede al sistema con le proprie credenziali e verifica le richieste pendenti, così da organizzarsi per evadere le consegne. Il magazziniere, periodicamente, rifornisce il magazzino, così che nuova merce risulti disponibile ai dipendenti della multinazionale. Il capo progetto, periodicamente, accede al sistema con le proprie credenziali e stampa un rapporto delle spese di articoli da ufficio così da verificare che non superi i limiti imposti per lo specifico progetto; il capo progetto può organizzare il rapporto raggruppando le spese per dipendente oppure per progetto (nel caso in cui egli sia a capo di più progetti). Devono essere previste tre diverse interfacce al sistema: un'interfaccia ad uso del dipendente, un'interfaccia ad uso del magazziniere, un'interfaccia ad uso del capo progetto. Il meccanismo di autenticazione al sistema deve essere unico.

La base di dati del sistema deve contenere almeno:

- 5 prodotti; 10 articoli; 3 categorie di prodotti; 3 produttori; 3 fornitori.
- 2 sedi della multinazionale con 1 magazzino per sede.
- 2 dipendenti; 2 magazzinieri; 2 capi progetto.

Indicazioni per lo svolgimento

Il sistema dovrà essere realizzato con un linguaggio di sviluppo orientato agli oggetti (preferibilmente Java); i dati dovranno essere memorizzati in una base dati relazionale (preferibilmente MySQL). Si dovrà adottare un processo di sviluppo agile (Scrum) e documentare il procedimento di lavoro adottato. L'elaborato software può essere svolto singolarmente o in gruppi di due studenti (due studenti è il numero massimo consentito, non saranno concesse deroghe per nessun motivo).

Due giorni solari prima dell'appello d'esame orale devono essere inviati al docente per email la documentazione di progetto in formato PDF e il software realizzato, in un unico file compresso in formato ZIP. I software consegnati saranno sottoposti a test di similarità, utilizzando strumenti automatici (tipo DIFF o più evoluti).

Il giorno dell'esame orale dovrà: (1) essere fornita la documentazione stampata, (2) eseguita una dimostrazione interattiva del sistema software. A ogni componente del gruppo, singolarmente, sarà richiesto di modificare il software utilizzando l'ambiente di sviluppo, implementando nuovi requisiti forniti dal docente al momento. La documentazione dovrà essere composta da un unico documento contenente: (1) descrizione dei requisiti del sistema, (2) analisi dei requisiti, (3) progettazione UML dell'architettura software, (4) progettazione concettuale e logica della base dati, (5) descrizione degli eventuali test eseguiti, (6) Scrum "sprint backlog" e "burndown chart".