

<p align="center">Corso di INGEGNERIA DEL SOFTWARE Corsi di Laurea in INGEGNERIA INFORMATICA e in INGEGNERIA DELL'AUTOMAZIONE A.A. 2014-2015</p> <p align="center">ESAME DEL 16 GIUGNO 2015 DURATA DELLA PROVA: 2,5 ORE</p>	<p align="center"><u>STUDENTE</u></p> <p>Cognome e Nome:</p> <p>Matricola:</p>
---	--

Prima Parte della Prova

1. Si consideri il seguente documento di descrizione informale dei requisiti:

Si vuole realizzare un'applicazione web e mobile per la gestione di un servizio di noleggio auto con conducente (NCC). Per essere abilitati al servizio, i conducenti devono fornire all'amministratore del sistema una copia della patente e i documenti di una o più auto messe a disposizione per il servizio di noleggio. L'amministratore del sistema registra e abilita i conducenti e memorizza le informazioni delle auto che i conducenti guidano. Per ciascun conducente, si vogliono memorizzare il numero di patente, anno di conseguimento e data di scadenza e se è abilitato al servizio di noleggio. Per le auto guidate dai conducenti, si vogliono memorizzare la targa, il modello, l'anno d'immatricolazione, la scadenza dell'assicurazione RCA e il segmento della vettura.

I conducenti abilitati possono inserire, modificare o rimuovere i turni nei quali sono disponibili. Per ciascun turno, i conducenti specificano le zone geografiche in cui prestano il servizio (città o provincia), il giorno e l'orario di inizio e di fine.

Per accedere al servizio di noleggio, tutti gli utenti devono loggarsi al sistema attraverso l'inserimento di username e password fornite al momento della registrazione. La procedura di registrazione consiste nell'inserimento dei dati personali quali nome, cognome, indirizzo, e-mail, numero di telefono, username e password.

Gli utenti loggati possono ricercare auto con conducente che siano disponibili per il noleggio: la ricerca si effettua specificando zona geografica, giorno, e ora. Dopo aver visualizzato la lista delle auto disponibili, gli utenti possono prenotare un'auto. La disponibilità di una vettura è data da: (1) la presenza di un turno che copre l'intervallo orario specificato dall'utente, (2) per la medesima vettura, l'assenza di sovrapposizioni con altre prenotazioni *attive* (ovvero, *non annullate*, secondo la definizione più avanti introdotta). Al momento della prenotazione l'utente può richiedere una stima del prezzo della corsa, specificando il numero di chilometri indicativi da percorrere: la stima viene calcolata col il tariffario in vigore. La prenotazione inizialmente si trova nello stato "PRENOTATA": fino a mezzora prima della corsa l'utente può richiedere al sistema di annullarla. L'annullamento viene registrato dal sistema, cambiando lo stato della prenotazione in "ANNULLATA". Se la corsa viene espletata, il conducente notifica il sistema del buon fine della prenotazione, specificando i chilometri effettivamente percorsi: in base a questi, il sistema calcola e registra il prezzo della corsa. Inoltre, se il cliente desidera pagare con carta di credito, all'atto della notifica il conducente richiede al sistema di eseguire la transazione finanziaria, interagendo con un sistema esterno di pagamento.

Per tale sistema, si realizzino in linguaggio UML:

- il diagramma dei casi d'uso

- il diagramma delle classi raffinato, che mostri anche attributi e responsabilità attribuite alle classi;

Suggerimento: si progetti applicando le buone norme dell'ingegneria del software, utilizzando in maniera opportuna le classi «boundary», «controller» e i design pattern.

- il diagramma di sequenza raffinato della funzionalità di ricerca di auto disponibili in base alla zona, giorno e ora.

Seconda Parte della Prova

Lo studente produca autonomamente i seguenti artefatti, da presentare sia in forma di relazione stampata sia in formato elettronico per la discussione alla successiva prova orale.

2. Si implementi in linguaggio Java la parte del sistema necessaria a realizzare la funzionalità **della ricerca auto disponibili in base a zona, giorno e ora**.

- Si realizzi con JDBC una classe DAO per una delle entità persistenti del sistema.

A corredo della relazione, lo studente dovrà consegnare per la discussione alla prova orale:

- I file del progetto (comprese le eventuali librerie – è consigliato l'uso di Eclipse per la gestione dei file del progetto). Come nome del progetto si indichi il proprio cognome seguito dal numero di matricola completo (ad esempio: ROSSI_N46000000)
- Una base di dati che permetta di esercitare la classe DAO implementata. È consigliato scegliere come DBMS di supporto H2 (<http://www.h2database.com/>) o Microsoft Access.

4. Si progetti ed esegua un insieme di casi di test black box per la funzionalità **della ricerca auto disponibili in base a zona, giorno e ora**, tenendo conto anche delle diverse pre-condizioni per tale funzionalità.
Si usi lo schema seguente per definire i casi di test.

T C	Descrizione Test Case	Classi di Equivalenza coperte	Precondizioni	Input	Output Attesi	Post- Condizioni Attese	Output Ottenuti	Post- Condizioni Ottenute	Esito

Lo studente dovrà consegnare:

- Un file di testo (formato .DOC, .RTF o .XLS) contenente la tabella dei casi di test progettati;
- Un Progetto di Test in JUnit contenente i casi di test necessari a testare la funzionalità richiesta;
- Esito dei test eseguiti con JUnit e opzionalmente i livelli di copertura del codice raggiunti.

- il *Class diagram* di progettazione di dettaglio del sistema; si trascuri in questo diagramma la gestione della persistenza, perché si suppone sia realizzata da un framework esterno, l'interfacciamento con gli utenti e le politiche di security (autenticazione con gli utenti);

il *Sequence diagram* di dettaglio per la realizzazione della funzionalità **della ricerca in base alla città di tutte le visite guidate che includono un'opzione che prevede l'utilizzo del tram**