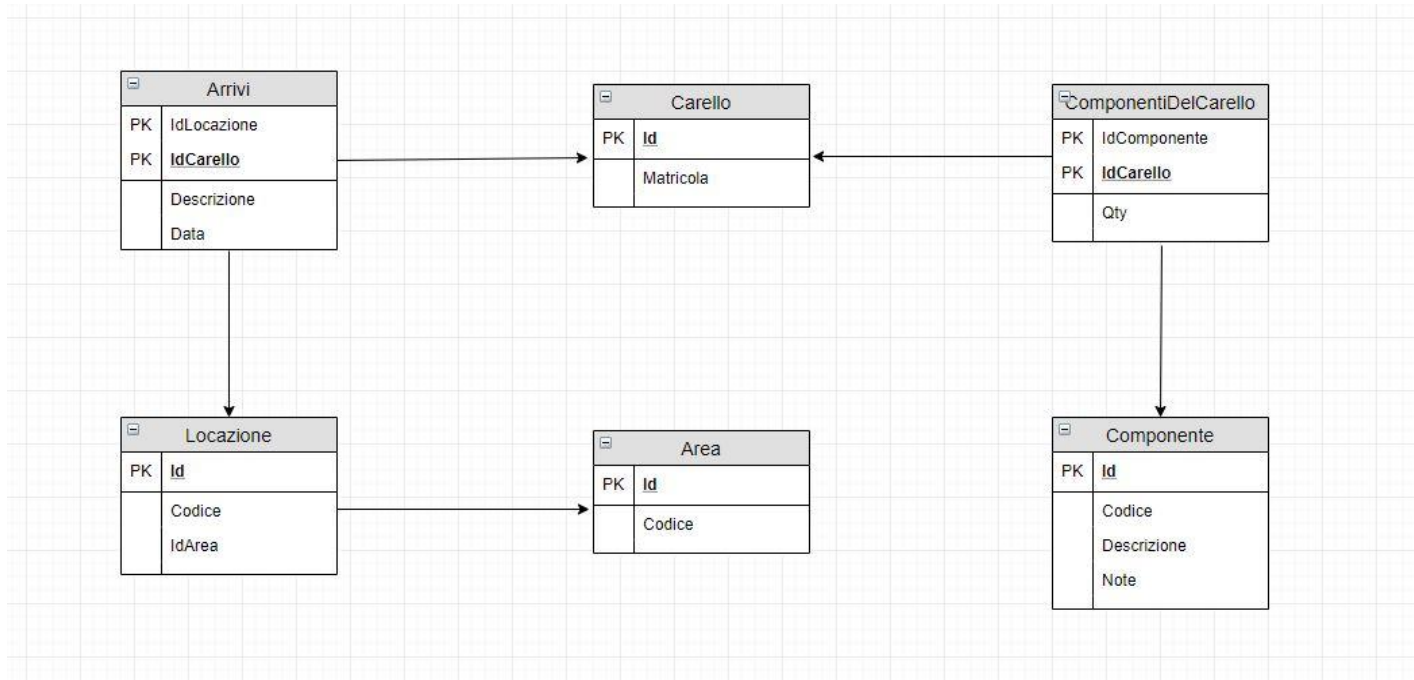


Esercizio: Identificazione merce in arrivo

Totale ore : 40

Progettazione database : 4 ore



I- Prima domanda (versione Modale): 12ore

Realizzare una anagrafica con la quale poter inserire/modificare/eliminare componenti di un carrello.

Ho usato l'entity framework core per generare il database e per questa problematica ho usato un'unica tabella "componente"

-Model: **Componente**

-Controller: **Componenti**

-View: **Componenti**

Su questa domanda la difficoltà è stata al livello di fare uscire la pagina di inserimento in un popup, la pagina **index** avendo come modello un **IEnumerable** non capivo come fare per avere 2 modelli in una pagina perché volevo usare "**asp-for**" per il data binding verso il controller e quindi dopo un po di ricerca ho scelto

come approccio quello di avere 3 pagine , **_ViewAll** per l'elenco delle componenti , **AddOrEdit** per inserimento o modifica di una componente e index in cui ho integrato la pagina _

Nella pagina **_Layout** che fa RENDER della pagina index precedentemente citato ho inserito una div con class="modal" per poterci inserire la pagina AddOrEdit in un popup per tale scopo ho creato una classe **Helper** che si trova in nella cartella **Other** che ha come scopo di trasformare la mia pagina AddOrEdit in una stringa e inserirla nella pagina _Layout e quindi il mio problema di integrazione di 2 modelli in una pagina è stato risolto.

II-Seconda domanda: 20ore

Realizzare una funzionalità (caso d'uso) che permetta di inserire un arrivo di un carrello e specificare tutti i componenti cui è caratterizzato e la posizione di stoccaggio in magazzino.

Per questa problematica sono andato con il presupposto che abbiamo già un elenco complete di tutte le Aree , per poter usare una dropdownlist nel mio Form , comunque ho anche creato un interfaccio per la gestione delle Aree nell'interfaccia delle Locazioni. Pensando che questa domanda fosse simile alla prima ho perso un po' di tempo ad impostare anche le funzionalità di modifiche e cancellazione di un arrivo mentre la problematica era solo al livello dell'inserimento quindi alla pagina /Arrivi/Index si ha anche la possibilità di modificare le informazioni dell'accettazione e alla pagina

/ComponentiDeiCarelli si può modificare le componenti del carrello inserito durante l'accettazione.

Le difficoltà incontrate a questo punto sono state:

-“Rchiesta Ajax” : questa volta ho deciso di non usare il data binding verso il controller ma di recuperare tutti i dati tramite Id , però un arrivo coinvolge **tutte** le tabelle del mio database quindi ho creato una classe DbViewArriviComponente per recuperare tutte le info che mi servivano per l'inserimento nel DbContext , questa classe contiene un array di CustomComponent che è una classe che ha come

attributi l'id della componente che si trova nel carello e la quantità, un CodiceLocazione che verrà inserito nella tabella Locazioni se e solo se non esiste già una locazione con lo stesso codice e lo stesso Areald , una matricola corrispondente alla matricola del carello che viene accettato e che verrà inserito nel database , un identificativo che corrisponde al codice identificativo dell'accettazione.

Ho passato quasi 5 ore su vari forum per risolvere il problema dell'errore 400 Bad request per colpa della **RequestVerificationToken**, poi il mio programma andava in errore 500 per altre 2 ore ho cercato qual'era il problema e finalmente ho capito che volevo inserire nella tabella ComponentiDeiCarelli una componente già presente nel carello e dato che questa tabella ha una chiave composta (carelloId e ComponenteId) ho messo una condizione di inserimento che andava ad incrementare la quantità se già presente nella tabella.

III-Terza domanda: 4ore

Realizzare una funzionalità (caso d'uso) che permetta di visualizzare l'elenco dei componenti presenti in magazzino con dettagli relativi al carrello di appartenenza e posizione di stoccaggio.

Per questa parte ho gestito tutto solo con il modello il database progettato apposto ho solo usato la funzione INCLUDE dell'Entity Framework per avere tutte le info che mi servivano e ho inserito il tutto in una tabella. Ho usato come modello di partenza ComponenteDelCarello e ho fatto i vari collegamenti per aver il risultato voluto.