# PROJECT 1 - WPF + SQL Server

### 1 - Colorize!

Finestra con colori salvabili su database.

A sinistra, elenco di colori salvati

A destra, dettaglio del colore selezionato, con controlli per settare RGB, HSL, …

Tasto Save -> Salva su Database (insert se nuovo, update se già salvato)

Tasto Refresh: ricarica lista da database

Tasto Undo: annullo le modifiche fatte al colore selezionato

### 2 - Photobook

Applicazione per salvare foto su database, caricandole tramite FileDialog

Tasti per convertirle in JPEG da PNG e viceversa

Applicazione di 2 semplici filtri a scelta

Schermata con statistiche sulle foto (quante di quale tipo, quante con filtri applicati, media delle dimensioni, dimensione massima e minima…)

In alto un pannello scrollabile orizzontalmente mostra le miniature delle foto

In basso la foto ingrandita con i filtri, tasti Save, Undo, ecc.

### 3 - Need for Speed - IRES edition

Videogame. Una macchina attraverso la città.

La città è un'immagine molto grande che va fatta scorrere sotto la macchina.

La macchina viene fatta muovere con i 4 tasti freccia, e viene fatta accelerare e decelerare con altri due tasti a scelta.

Se la macchina va contro una casa o un muro, si rompe, non può più muoversi e cambia la sua immagine.

Tasto Restart per ricominciare.

Ci sono 3-4 mappe salvate su database, selezionabili.

### 4 - Shops warehouse

Tipico gestionale per compagnia di vendita.

Ci sono un elenco di negozi, un elenco di dipendenti, una lista di scontrini emessi un certo giorno da un certo dipendente, ogni scontrino ha una lista di voci, ogni voce è correlata a un certo scontrino e un certo prodotto.

Di ogni prodotto si sa quanti pezzi ci sono nel magazzino; quando viene emesso uno scontrino bisogna controllare che effettivamente ci fossero abbastanza pezzi di quel prodotto in magazzino, e bisogna scalare la quantità.

Realizzare una serie di schermate per gestire le operazioni di CRUD su Dipendenti, Scontrini, Prodotti in Magazzino.

### 5 - Arena

Dall'esercizio fatto un mese fa.

Stesse regole di combattimento, ma mostrate graficamente.

La lista di giocatori è caricata da database.

Vengono disposti in cerchio

Ogni tipo di giocatore ha un'icona diversa (3 immagini: guerrieri, maghi, orchi)

Ad ogni turno un giocatore avanza verso l'avversario, lo colpisce e torna alla sua posizione.

Il colpo ha differenti effetti grafici in base al tipo di giocatori

Se un giocatore muore, l'immagine cambia e nei turni dopo non interviene più.

### 6 - Dama cinese

Scacchiera con struttura triangolare e I pedoni inizialmente disposti come da regolamento.

Posso selezionare un mio pedone, nel mio turno, e muoverlo (in una posizione consentita), e mangiare un pedone avversario

L'applicazione deve riconoscere se qualcuno ha vinto.

Una partita può essere salvata su database e caricata più tardi nella situazione in cui era.

Le vincite sono salvate su database, e c'è una schermata che mostra lo storico delle vincite.

### 7 - Labirinti

Al click di un pulsante si genera un labirinto casuale.

Il programma deve anche dire (con algoritmo ricorsivo) se il labirinto generato ha una via d'uscita o no.

I labirinti si possono salvare su database e ricaricare.

A sinistra: lista di labirinti (a cui si può dare un nome).

A destra dettaglio del labirinto.

### 8 - Mechanical components

Ogni motore è fatto da un insieme di componenti, a loro volta fatti da un insieme di componenti. Componenti di tre tipi: con una lista di figli, con un figlio unico, senza figli.

I componenti sono mostrati in un TreeView a sinistra. In base al tipo devono visualizzare un'icona diversa.

Tutti i componenti sono memorizzati su database in modo relazionale.

Parto con un treeview con la lista di motori, senza figli caricati. La prima volta che clicco per espandere un nodo, i figli sono caricati dal database e mostrati, e da quel momento tenuti in memoria.

Con un tasto "Refresh" ricarico i figli del componente selezionato (se non ho componente selezionato, il pulsante è disabilitato).

A destra ho il dettaglio del componente selezionato sul TreeView di sinistra, e a seconda del tipo di componente posso settare alcune proprietà (da vedere…), oppure aggiungere un figlio solo o aggiungere più figli.

### 9 - Dati di borsa

Ho N aziende in borsa, ognuna ha una timeline di valori (cioè il titolo valeva 24 € alle 9:00, poi 25 alle 9:30, …).

Devo realizzare un grafico che mostri i vari andamenti di borsa con colori diversi.

Ho l'elenco di aziende sulla destra in forma di legenda di checkbox; se deseleziono un'azienda, il suo grafico scompare.

Il grafico deve adattare le sue dimensioni, il suo range, in base a quali andamenti sono mostrati.

Il che significa anche riadattare la griglia, che deve mostare sull'asse X e Y un tot di valori a intervalli regolari.

Su un'altra schermata aggiungo valori alla timeline della singola azienda.

I dati sono salvati su database.

### 10 - Disegno 3D

Applicazione per disegnare poliedri, da mostrare in 3D.

A sinistra, elenco di poliedri.

Al centro, pannello 3D.

A destra, elenco di linee che compongono il poliedro (ognuna con i suoi due estremi, espressi come terza di coordinate 3D).

Man mano che aggiungo linee nella lista di destra, al centro vedo linee grafiche comparire. Il pannello centrale deve essere ruotabile con Pan 3D.

L'elenco di poliedri è salvato su database, e ricaricato all'inizio dell'applicazione.

### 11 - Ricette

Applicazione per gestire un'anagrafica di ricette da cucina.

Ogni ricetta ha un elenco di ingredienti, e un elenco di azioni da fare.

Per ogni ingrediente o voce l'utente può caricare un'immagine.

Le ricette sono di tipo diverso ("dolce", "pasta", "drink", …). In base al tipo, solo alcuni ingredienti e azioni possono essere fatte (un impasto per un dolce non può essere "shakerato").

Un ingrediente può essere di vari tipi (tipo che si misura in grammi, o in cucchiai, o in lunghezza…).

Un'azione ha una descrizione presa da un elenco fisso ("mescolare", "accendere il fuoco", …). Vicino, fino a 3

### 12 - Home Banking

Gestire un'anagrafica di persone e investimenti

Gli investimenti sono di diverso tipo (mutuo, investimento di puro guadagno, …).

Per ogni investimento ci sono parametri diversi (mutuo e prestito richiedono un garante se i beni posseduti dalla persona non sono sufficienti o il contratto di lavoro non è di tipo indeterminato

### 13 - Nerd Student

Uno studente vuole gestire il proprio andamento a scuola, con l'elenco di materie; per ogni materia, l'elenco di interrogazioni o compiti scritti, comprese le consegne e il voto; per ogni materia, anche l'elenco di compiti per casa, fatti da un elenco di istruzioni (studiare da pagina X a pagina Y, fare esercizi 123-125 a pag. 42, ripassare con compagno di classe Z, …).