

ANALISI DEL PROGETTO

SCOPO DEL DOCUMENTO

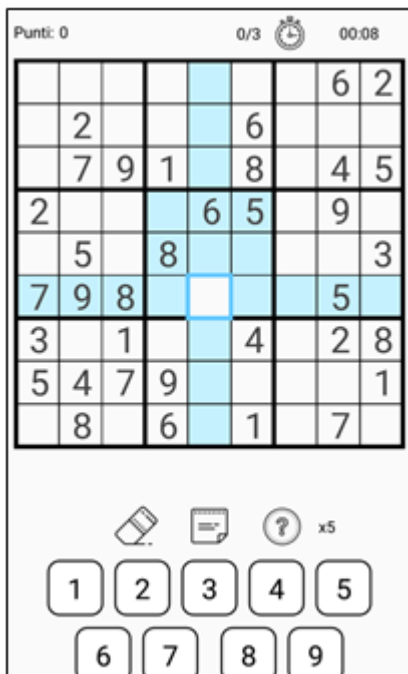
Nell'analisi del progetto verranno messe alla luce le scelte implementative riguardanti i punti chiave delle specifiche del progetto "Sudoku". Gli argomenti trattati saranno:

- Logica di generazione delle partite
- Gestione interfaccia utente e svolgimento del gioco
- Database dei risultati
- Logica di risoluzione del sudoku e generazione mossa intelligente

Logica di generazione delle partite

All'atto della richiesta di una nuova partita, questa verrà generata tramite un API web all'indirizzo <https://sugoku.herokuapp.com/board?difficulty=%s>, specificando la difficoltà come parametro di tipo GET. All'interno dell'applicazione, questa responsabilità è assegnata alla libreria Volley. Il Json, risultato della richiesta HTTP, viene interpretato e trasformato in un formato accettato dall'applicazione, in modo tale da poter generare correttamente lo stato iniziale di una tabella di sudoku. Nel caso in cui la richiesta non dovesse andare a buon fine, la partita viene generata casualmente estraendola da uno storage locale. In particolare, vengono memorizzati negli Asset dell'applicazione dei file di testo contenenti 60 partite (20 per difficoltà).

Gestione interfaccia utente e svolgimento del gioco



L'interfaccia utente di una partita di sudoku rispecchia la più tipica modalità di gioco, attualmente utilizzata in tutte le applicazioni commerciali. È possibile vedere in alto il proprio punteggio, il numero di errori commessi, un timer accoppiato all'animazione dell'orologio che ruota in accordo allo scorrere del tempo. L'interazione con la griglia è facilitata grazie al cursore, cioè l'oggetto grafico (in figura di colore blu) che evidenzia la cella selezionata e tutte le celle con cui questa può andare in conflitto in funzione delle regole di inserimento dei numeri nel sudoku. I 3 bottoni presenti sotto la tabella svolgono funzioni intuitivamente deducibili dalla loro forma stessa: è possibile cancellare le note dalla cella selezionata, passare dalla modalità inserimento note alla modalità inserimento numeri (l'utente verrà aiutato in questo procedimento grazie al cambiamento di colore del rispettivo bottone), chiedere un aiuto al gioco, con il numero di aiuti rimanenti scritti di fianco. Ogni click è accompagnato da un feedback sonoro per rendere l'esperienza di gioco più gradevole. Inoltre, durante l'intera

partita, l'utente ascolterà una musica di sottofondo la quale può essere disattivata dalle impostazioni o cambiata con un proprio file .wav.

Alla pressione di un numero (rappresentati dai bottoni da 1 a 9) verrà inserito il corrispettivo valore all'interno della cella, se corretto, seguito da un suono appropriato. Alternativamente, verrà segnalato l'errore tramite un feedback sonoro e tramite all'illuminazione di colore rosso della cella

in cui è avvenuto l'errore (fondamentale in caso l'utente avesse l'audio disattivato o troppo basso). Quando un valore è inserito correttamente in una cella vengono rimosse automaticamente le note obsolete, cioè le note inserite dall'utente che non sono più valide per via del nuovo valore che compone la matrice attuale. Al termine di una partita vinta sarà possibile inserire il proprio nome utente per memorizzare il punteggio in un database locale.

Database dei risultati

I risultati sono memorizzati su di un database locale. È stata utilizzata la libreria Room per la gestione del database SQLite.

La classifica, mostrando i risultati in ordine discendente nei punti, permette la cancellazione degli score non desiderati.

Logica di risoluzione del sudoku e generazione mossa intelligente

Due operazioni fondamentali durante la partita di sudoku sono: verificare se la mossa dell'utente è corretta (quindi stabilire se un numero è corretto in una cella) e fornire all'utente un auto-completamento di una cella (operazione successiva alla richiesta di aiuto tramite l'apposito bottone).

Per fornire all'utente queste due funzionalità, l'applicazione calcola la soluzione del sudoku all'inizio della partita e la memorizza.

L'algoritmo di risoluzione del Sudoku è stato sviluppato in modo tale da evitare un approccio brute force quando è possibile, in modo da offrire prestazioni migliori ed evitare latenze nell'inizio partita.

L'algoritmo è quindi suddiviso in due step:

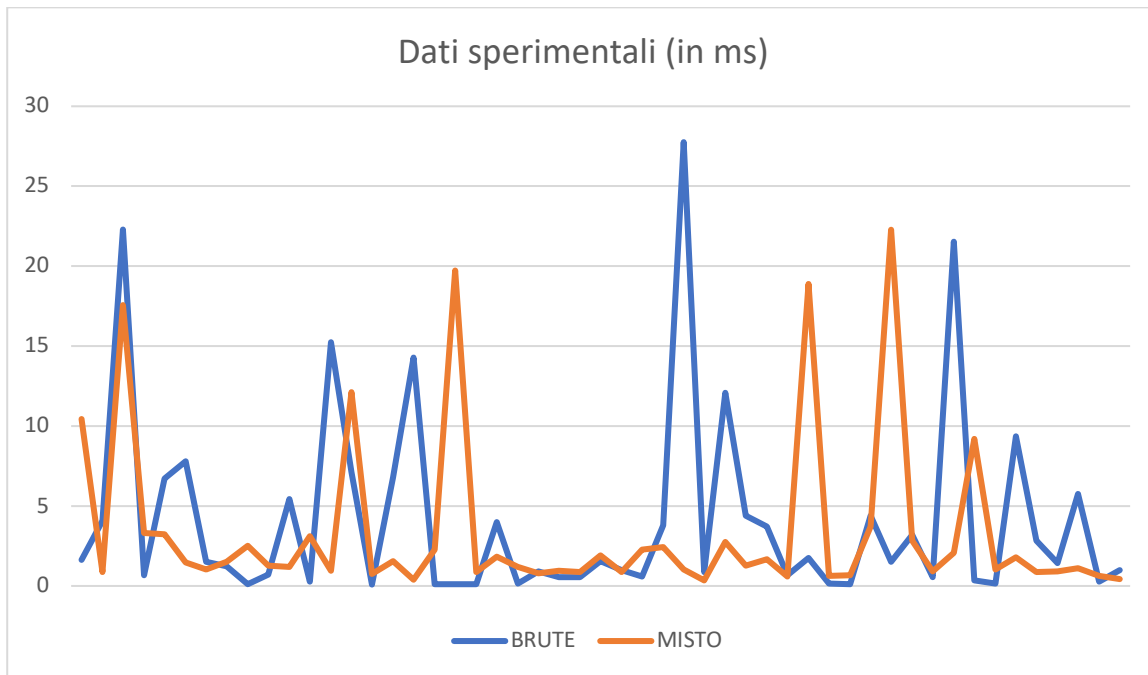
1. **Deduttivo:** Identificare i valori che si possono sicuramente dedurre, discorso approfondito nella presentazione attraverso un teorema e delle proposizioni;
2. **Brute-force:** Ipotesizzare una possibile soluzione e vedere se crea una contraddizione. Nel caso contraddittorio prenderemo per errata l'ipotesi e ne tenteremo un'altra.

Lo sviluppo tiene anche conto che la web-api utilizzata restituisca sudoku almeno 8 valori inseriti (condizione sempre verificata).

A questo punto, la verifica della correttezza di un valore in una cella è istantanea: basta confrontare il valore con la matrice risolta. Invece, alla richiesta di un aiuto, viene auto-completata una cella casuale presente nella riga o colonna o quadrato con meno valori mancanti. L'obiettivo è quindi di fornire all'utente un aiuto che sarebbe stato il meno possibile difficile da trovare autonomamente dall'utente stesso.

L'algoritmo infine sviluppato permette di avere, ad un piccolo costo iniziale medio di 10 ms, un'esperienza utente estremamente fluida in quanto le singole verifiche non necessitano di nessun ulteriore calcolo ma di un unico confronto.

Dati sperimentali



Deduciamo dai dati sperimentali che mediamente l'approccio misto (**deduttivo + brute**) sia sempre migliore o uguale all'approccio esclusivamente brute-force, che è quindi da non preferire. Per questa ragione è opportuno, prima di risolvere il sudoku tramite brute-force, calcolare in maniera deduttiva le celle con valori obbligati.

Profilazione

Dai dati della profilazione notiamo come l'applicazione ha un basso consumo di batteria e CPU. L'operazione più onerosa è la risoluzione del sudoku, la quale impiega risorse in maniera moderata per un tempo molto breve (nell'ordine delle decine di ms). Si nota un picco dell'utilizzo della rete durante il download della tabella di sudoku dal server.

