*Piccinin Luca* *5CIA,1/2/2021*

**Android App Development**

**Technical Specification**

**Quiz World**

**Abstract**

La mia applicazione principalmente non è affatto complicata come idea in sé, il concetto di base è quello di “fare” dei quiz su diversi e vari argomenti riguardanti qualunque cosa. Può essere facilmente usata per imparare cose nuove, ad esempio se si utilizzano quiz di varie materie scolastiche. Il motivo principale è che non avevo la minima idea di che applicazione progettare, finchè vedendo i miei genitori giocare ai dei giochi simili non mi è venuta l’idea di provarne a fare una.

**General Technical Data**

Choose the features and components included in your app:

Nell’app in questione, sono presenti quattro diverse activities.

|  |  |  |
| --- | --- | --- |
| MainActivity | Quest’activity viene mostrata la prima parte dell’app, ovvero quella in cui si possono andare a leggere le regole per avere un’ampia spiegazione sul suo funzionamento, si va anche a decidere la lingua di base della nostra app, e in seguito all’inserimento del proprio nome utente si può accedere all’activity successiva |  |
| SecondActivity | Quest’activity invece si va a visualizzare attraverso uno Spinner la lista completa di tutti i possibili argomenti presenti per la lingua selezionata in precedenza, in seguito alla scelta dell’argomento e alla successiva pressione del bottone per iniziare il quiz si verrà mandati all’effettivo utilizzo dell’app |  |
| ThirdActivity | Quest’activity è quella principale della nostra app, in quanto in questa verranno mostrate le domande relative all’argomento scelto e in una EditText posta immediatamente sotto si potrà inserire la relativa risposta, sempre che la si conosca, e alla successiva pressione del bottone di invio della risposta si passerà immediatamente alla domanda successiva fino al termine di quest’ultime e quindi al completamento del relativo quiz. Inoltre in quest’activity vi è presente anche una progressBar che andrà a mostrare al giocatore la progressione delle domande dando così la possibilità di dedurre quante ne manchino al completamento |  |
| FourthActivity | Quest’activity è l’ultima presente nell’app ed è quella che riveste il ruolo di mostrare i relativi risultati al giocatore mostrando l’argomento scelto, il suo nome utente e il numero di risposte che sono state date corrette, nonché il numero di domande totali che sono state poste. Qui verrà anche mostrato un bottone che consente di tornare alla MainActivity e in seguito di iniziare un nuovo quiz se il giocatore ne ha voglia |  |

* Content Provider

L’unica tipologia di Content Provider da me implementata nell’app sono gli Intent in quanto usati ampiamente per passare i dati tra le varie activity all’interno dell’app.

* Intent

|  |  |
| --- | --- |
| Principalmente tra le varie activity vengono passati quasi sempre gli stessi Intent. Dalla prima activity alla seconda ad esempio attraverso l’Intent vengono passati il nome utente inserito e la lingua che è stata scelta e che verrà usata per tutto il resto dell’app |  |
| Dalla seconda activity alla terza vengono passati nuovamente i parametri del nome utente e della lingua scelta, ma inoltre viene aggiunto anche il valore contenente l’argomento scelto che nella successiva activity verrà usato per mostrare il vero quiz |  |
| Dalla seconda activity alla terza vengono passati un’altra volta i parametri che contengono il nome utente, la lingua e anche l’argomento. Inoltre si vanno ad aggiungere i risultati del quiz appena svolto ovvero il numero totale delle domande che sono state poste nonché il numero di risposte corrette che sono state date dal giocatore |  |
| Nell’ultima activity, ovvero quella usata per mostrare i risultati viene usato un Intent per tornare alla MainActivity. In questo caso però non viene passato nessun valore al passaggio alla suddetta activity in quanto quest’ultima è quella che per prima darà dei valori da passare in input e per questo non ne richiede necessariamente da altre per poter funzionare correttamente |  |

* SQLite database engine

Nel mio progetto ho ultimamente implementato un database SQLite per l’implementazione di una classifica più completa nella quarta activity. In questo database vengono utilizzati tutti i dati raccolti nelle varie applicazioni, ovvero il nome del giocatore, la lingua, l’argomento del quiz, le domande totali e infine quelle che l’utente ha risposto corrette. In questo database si vanno a raccogliere tutti questi risultati per poi nell’activity finale avere la possibilità di visualizzare tutti i risultati dei quiz svolti sul dispositivo che si sta usando, avendo così la possibilità di confrontarsi con i risultati di quiz già completati per vedere se si è riusciti a migliorare.

* Multiple device layout and resolution support

I dispositivi in cui quest’app possiede un layout e una risoluzione adeguata e supporta dal codice scritto è quella degli smartphone, con dimensioni simili al Pixel 2 e non inferiori. In quanto in presenza di dispositivi con dimensioni piccole o con layout orizzontali il layout dell’app subirà modifiche non adeguate al funzionamento. Ad esempio ci sarà la scomparsa di alcune features o la sovrapposizione di alcune andando ad impedire il corretto nonché il pieno funzionamento dell’app.

* Other (specify)

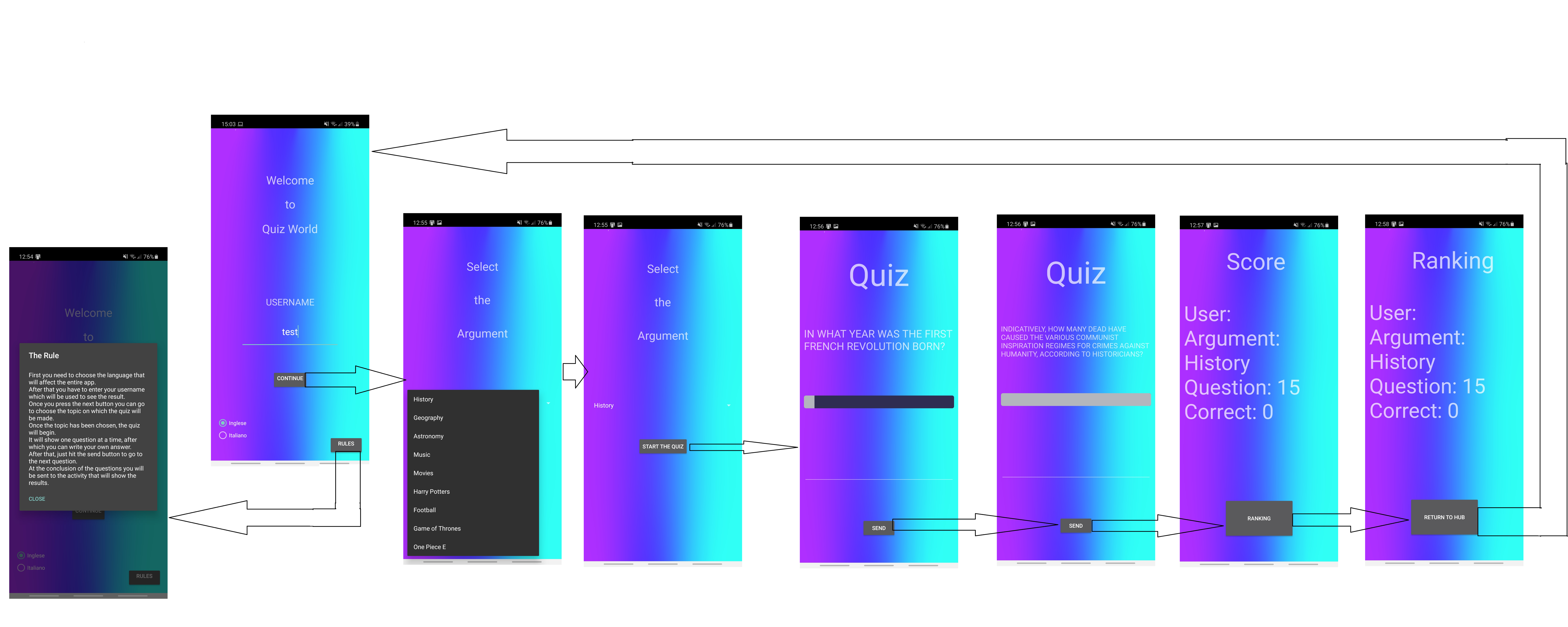
Nella mia applicazione non vi è la presenta di nessun DBMS esterno, ma viceversa ho implementato l’uso degli assets che permettono l’accesso a file di risorse non elaborate di un'applicazione.

**Key Features**

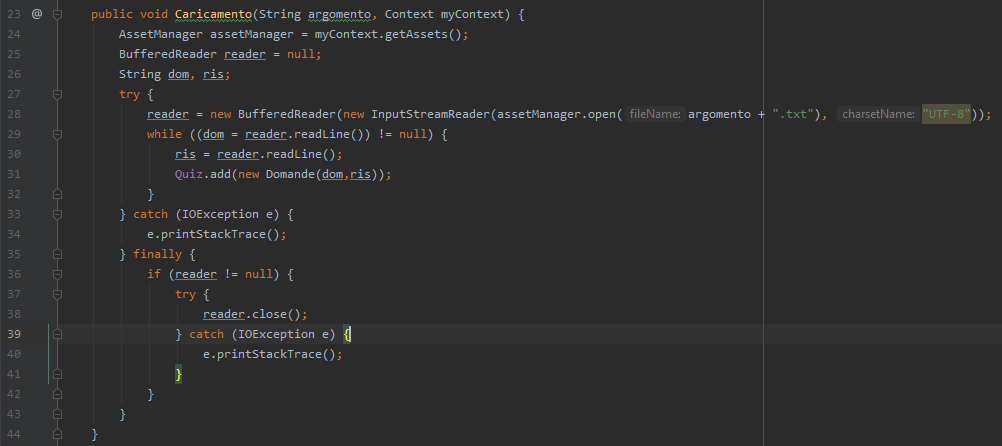
A mio parere ciò che rende a mia applicazione buona o che potrebbe diventare buona, è il fatto principale che potrebbe essere usata per migliorare l’apprendimento nel caso in cui si vadano ad implementare quiz creati per l’uso scolastico. Basti pensare che si possano usare quiz comprendenti argomenti di storia, geografia, matematica, ecc..., tutti questi possono essere usati per migliorare o per verificare se si possiede la conoscenza di certi aspetti nei vari argomenti proposti.

**App structure overview**

***Wireframe***



**Code fragments**



Quello nell’immagine è un estratto della classe Quiz, la classe che implementa i metodi per la lettura dei file di testo dei vari argomenti caricando il loro contenuto in una lista di Domande (altra classe usata per unire domande e risposte).

Iniziamo subito parlando di cosa sono gli Asset:

Android offre un'altra directory in cui puoi conservare i file che saranno inclusi nel pacchetto. Questa directory è chiamata Assets. La differenza tra Res e Assets è che Android non genera ID per il contenuto delle risorse. Nel codice in sé non è necessario specificare il percorso, basta solamente il nome relativo del file da usare.

I file contenuti nella cartella Assets possono solamente essere letti e perciò non possono essere usati per scrivere al loro interno.

Per prima cosa bisogna aprire i file all’interno della nostra activity:

-AssetManager assetManager = getAssets();

AssetManager permette di aprire i file dentro la cartella assets tramite il metodo open() che ritorna un InputStreamReader, quindi si può andare a creare un elemento BufferedReader che vada a contenere al suo interno l’elemento InputStreamReader ritornato dall’AssetManager

- new BufferedReader(new InputStreamReader(assetManager.open(NomeFile), "UTF-8"));

La parte di codice che viene inserita dopo il nome del file, in questo caso UTF-8 è uno standard utilizzato affinché durante la lettura dal file l’AssetManager sappia quale set di caratteri deve andare ad utilizzare. Questo set di caratteri va a coprire praticamente tutti i caratteri e i simboli de mondo.

**Development**

https://github.com/luca-piccinin/Quiz-World.git

Target API level: 29 / 30

Minimum API level: 26

IDE: Android Studio whit Java

Man-hours: 20 / 25

***Further development***

La mia applicazione potrebbe essere migliorata col tempo, ad esempio aggiungendo la possibilità di creare da soli dei quiz, non per forza in entrambe le lingue, o aggiungendo la possibilità di aumentare il numero di domande per un dato argomento.

***Self-rating***

Alla mia applicazione darei all’incirca 2,5/3 stelle dato che possono essere implementate diverse funzionalità per poterla migliorare sia esteticamente sia nel suo stesso funzionamento.

**References**

Per creare quest’app sono stati usati diversi materiali esterni ed alcuni interni:

* I vecchi progetti delle app svolte;
* Gli appunti caricati dal prof. Camilotti;
* La teoria caricate dal prof. Camilotti;
* Alcune pagine di Stack Overflow;
* Alcune pagine di Html.it (molto simile alle precedenti pagine);
* Dei miei compagni di classe per aiutarmi a risolvere alcuni problemi riscontrati;
* Diversi siti per riuscire a creare i quiz dei diversi argomenti presenti;
* Google traduttore per tradurre i quiz in inglese.