All’interno di questo file, vengono spiegate e mostrate alcune modifiche apportabili all’applicazione ExploRL, un’applicazione di supporto per l’attività “Impara dagli errori, sperimenta il Reinforcement Learning” presente al Muse – Museo delle Scienze di Trento.

Il link dell’attività è: <https://ilmuseperlascuola.muse.it/attivita-didattica/reinforcemente-learnig/>

Questo file si divide in più sezioni, ognuna riguardante un tutorial per una specifica modifica possibile.

In caso di problemi o necessità di ulteriori informazioni, scrivere una mail a [lorenzo.dambro@gmail.com](mailto:lorenzo.dambro@gmail.com), mettendo come oggetto “Problemi mantenimento app ExploRL”.

La repo ufficiale dell’applicazione è reperibile dal link: <https://github.com/lorenzo2330/MUSE_ExploRL>

I tutorial sono aggiornati a febbraio 2025, sono stati realizzati con Android Studio Koala 2024.1.2, e fanno riferimento alla versione 1.0.1 dell’applicazione.

I tutorial presenti all’interno di questo file sono:

* Generare QR-code [fatto]
* Importare delle foto [fatto]
* Modificare parametri energia e timer [fatto]
* Modificare struttura animale [fatto]
* Modificare la lista di animali [fatto]
* Realizzare e caricare una nuova versione [fatto]

Infine, viene fornita una sezione contenente tutti i QR-code, utile per il debug.

# Generare QR-code

Per generare un QR-code, è necessario andare su un qualunque sito in grado di svolgere tale compito. L’unico vincolo è che, una volta scansionato il QR-code, esso fornisca un semplice testo, equivalente con quanto riportato nella sezione “normalName” della struttura dell’animale (mostrata nella relativa sezione “Modificare struttura animale”), compresi spazi e lettere maiuscole/minuscole.

Tutti i QR-code attualmente presenti all’interno dell’applicazione sono stati realizzati tramite il sito <https://qrplanet.com/designer-qr-code-with-embeded-logo#text>, il quale permette la realizzazione permanente (anche sottoforma di semplice testo) di QR-code in formato .png, .svg o .pdf (ai fini dell’utilizzo all’interno dell’applicazione, il formato del QR-code è equivalente).

# Importare delle foto nell’applicazione

Per importare una foto all’interno della struttura dell’applicazione, vi sono vari modi, il più semplice consiste nel salvarsi la foto in locale, copiarla (con un banale CTRL+C), selezionare la cartella MUSE\_ExploRL>images e incollare all’interno la nuova foto.

Nota: se si sta aggiungendo o modificando la foto di un animale, assegnare un nome coerente con quanto riportato nella struttura dell’animale (vedi paragrafi “Modificare struttura animale” e “Modificare lista animali”).

# Modificare parametri energia o timer

Per l’energia:

* Aprire il file MUSE\_ExploRL > lib > providers > energy\_provider.dart
* Modificare, nella prima riga della classe, il parametro maxEnergy

Per il timer:

* Aprire il file MUSE\_ExploRL > lib > providers > game\_provider.dart
* Modificare le righe relative a minuti e secondi



# Modificare struttura animale

Andare nel file MUSE\_ExploRL > lib > models > exhibit.dart

Inserire un nuovo dato all’interno della classe ed eventualmente anche all’interno del costruttore (quello con i vari required).

Immagine che contiene testo, schermata, software, Carattere

Descrizione generata automaticamente

NOTA: una volta effettuata la modifica, sarà necessario applicarla a mano a tutti gli exhibit già esistenti (vedi paragrafo “Modificare lista degli animali”)

# Modificare lista degli animali

Andare nel file MUSE\_ExploRL > lib > models > exhibit\_list.dart

Se si vuole modificare un valore di un animale già presente:

* Cercare la struttura relativa all’animale del quale si vuole modificare un valore
* Modificare il valore, facendo attenzione a ciò che potrebbe comportare tale modifica (ad esempio, se si sta modificando l’immagine o il nome, aggiornare anche il nome dell’immagine o il QR-code generato)

Nota: se si modifica il normalName, è necessario modificarlo anche nelle funzioni getExhibitByName e scannedExhibit (sempre all’interno del file MUSE\_ExploRL > lib > models > exhibit\_list.dart).

Se si vuole aggiungere un animale:

* Utilizzare il template già presente
* Rimuovere i simboli di commento /\* e \*/
* Assegnare i parametri

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

Se si vogliono aggiungere o rimuovere i “vicini” degli animali

* Scorrere in basso fino alla funzione setNeighbors()
* Modificare i vicini dell’animale desiderato

Immagine che contiene testo, Carattere, schermata

Descrizione generata automaticamente

Se si vogliono modificare gli animali di partenza, di fine o di tutorial:

* Scorrere in basso fino alle tre variabili: startingExhibit, winnerExhibit e tutorialExhibit
* Sostituire gli animali che si vogliono modificare

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

# Realizzare e caricare una nuova versione

Dopo aver apportato modifiche significative, rendendo quindi necessaria la pubblicazione di una nuova versione dell’applicazione, è necessario andare nel file MUSE\_ExploRL > pubspec.yaml, cercare la riga “version”, sostituire il numero di versione prima del + e incrementare il numero di versione dopo il +. È importante che quest’ultimo numero sia sempre maggiore del precedente.

Immagine che contiene testo, Carattere, schermata

Descrizione generata automaticamente

A questo punto è necessario aggiornare le dipendenze digitando nel terminale: flutter pub get

Il proseguimento si divide lato Android e lato iOS

## Android

Sempre da terminale, digitare flutter build apk –release, verrà generato un file in MUSE\_ExploRL/build/app/outputs/bundle/release/app-release.aab

A questo punto, è necessario andare sul sito GooglePlayConsole (<https://play.google.com/console/u/0/signup>) ed accedervi con l’account sviluppatore.

Dalla dashboard basterà selezionare l’applicazione MUSE\_ExploRL > Testa e rilascia > Produzione e selezionare il pulsante in alto a destra con scritto “Crea nuova release”

Nella schermata che si apre, caricare il file appBundle generato precedentemente e compilare tutti i campi presenti, una volta fatto il tutto, dopo un po’ di tempo, necessario per far si che lo staff di Google approvi l’aggiornamento, la nuova versione sarà caricata sul PlayStore.

## iOS

Andare sul sito AppStoreConnect (<https://appstoreconnect.apple.com>) ed entrare con le proprie credenziali da sviluppatore. Tra le varie applicazioni, selezionare MUSE ExploRL e, nel menu che si aprirà, cliccare sul + azzurro sotto la scritta MUSE ExploRL. Inserire il numero di versione e procedere.

NOTA: tutte le seguenti operazioni dovranno essere eseguite solo su un dispositivo Mac.

Aprire xCode e aprire il file MUSE\_ExploRL > ios > Runner.xcworkspace

Aggiungere il proprio account sviluppatore a xCode (xCode > Settings > Accounts), inserendolo o aggiornandolo (Download manual profiles) qualora non fosse presente. Assicurarsi inoltre di avere i permessi di develop per applicazioni iOS (vengono gestiti dall’amministratore dell’account).

Tornando su xCode, selezionare la voce Runner sotto la sezione TARGETS, a questo punto cliccare sulla sezione Signin & Capabilities e selezionare il proprio account sviluppatore alla voce Team.

A questo punto caricare una nuova build dell’applicazione cliccando su Product (nella barra superiore del Mac) > Archive, dopo qualche secondo si genererà una build. Successivamente cliccare su Distribute App > App Store Connect e cliccate Distribute.

Al termine del caricamento, tornare sul sito AppStoreConnect (<https://appstoreconnect.apple.com>) e sotto la sezione TestFlight, dopo qualche minuto, comparirà la nuova build. A questo punto compilare eventuali campi richiesti e salvare la versione, rendendola pronta per il rilasci, dopo una supervisione da parte di Apple.

# Schermate QR per agevolare il testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Immagine che contiene modello, quadrato, Simmetria, arte  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, pixel  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, Rettangolo  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, pixel, design  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, pixel  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, pixel, design  Descrizione generata automaticamente |
| Capibara | Cervo nobile | Dingo | Formichiere  gigante | Lemure catta | Leone africano |
| Immagine che contiene modello, quadrato, pixel, parole crociate  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, pixel, design  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, Rettangolo  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, pixel  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, pixel, Simmetria  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, pixel  Descrizione generata automaticamente |
| Leone marino sudamericano | Lince eurasiatica | Lupo grigio | Opossum della Virginia | Oritteropo | Ornitorinco |
| Immagine che contiene modello, quadrato, pixel, design  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, Rettangolo  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, arte  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, Rettangolo  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, Simmetria, arte  Descrizione generata automaticamente | Immagine che contiene modello, quadrato, pixel, design  Descrizione generata automaticamente |
| Otaria orsina  del Capo | Riccio comune | Springbok | Tasso | Tricheco | Wallaby dal collo rosso |