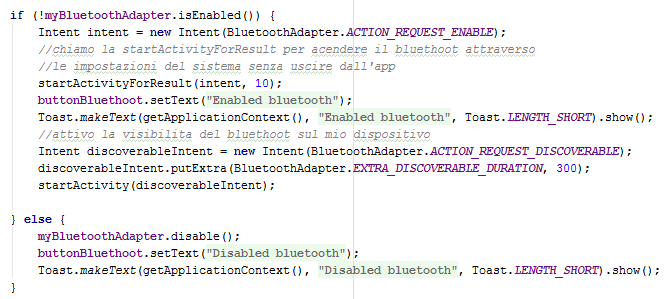
SpaceBattle è un’app che permette a due utenti di sfidarsi via Bluetooth ad una versione spaziale del gioco “Battaglia navale”, ispirata al tema di Star Wars.

* Senza Bluetooth non si gioca!

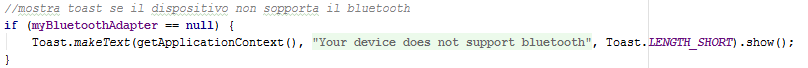
Lo scambio di informazioni necessarie a far procedere la partita avviene via Bluetooth, quindi è necessario che l’utente attivi il Bluetooth del proprio dispositivo prima di iniziare la partita, rendendolo scopribile per 300 secondi durante la scansione per la ricerca degli eventuali avversari.



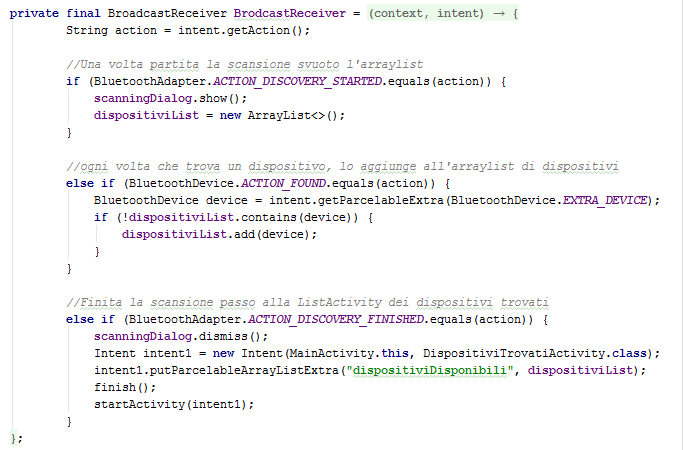
Il bottone collegato all’attivazione del Bluetooth, informa il giocatore sullo stato di attività o inattività del BluetoothAdapter, in modo da rendere più accessibile l’utilizzo. Per esempio:



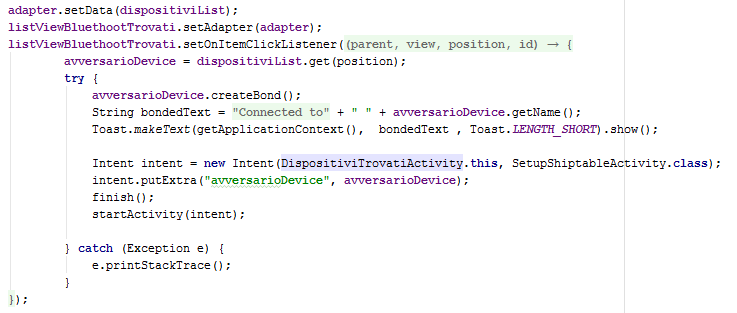
Nel caso il dispositivo non supporti il Bluetooth, rendendo impossibile l’utilizzo dell’applicazione, appare un Toast che lo comunica all’utente.



Un BroadcastReceiver si occupa di aggiungere ad un ArrayList di Devices ogni dispositivo localizzato durante la scansione e anche di mostrare un Dialog, che mette in attesa l’utente, fino alla fine della ricerca.

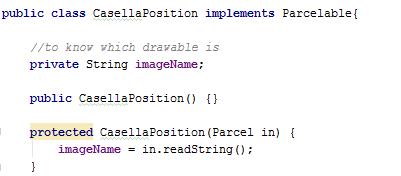
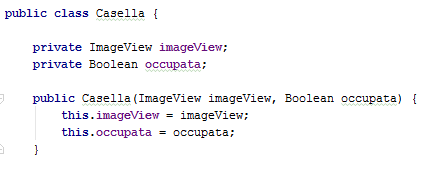


Il nome e l’indirizzo dei vari dispositivi trovati vengono mostrati in una customize ListView. Cliccando uno degli item della listView, si seleziona l’avversario con cui giocare, invocando il metodo createBond() che collega i due dispositivi.

Se non è stato trovato nessun dispositivo durante la scansione, il giocatore viene avvertito con Dialog che lo riporta alla pagina iniziale.

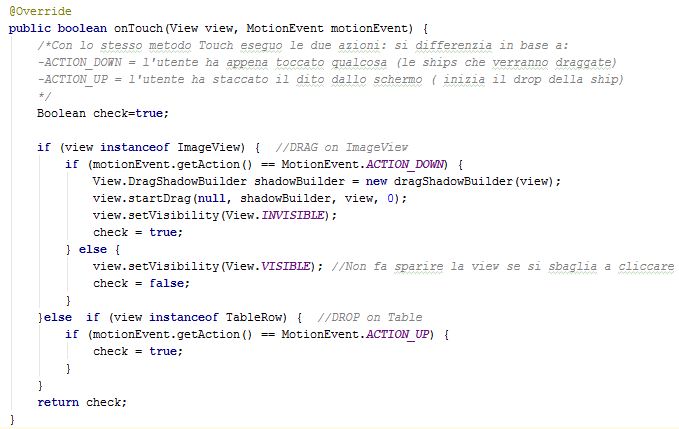
* Posizionare le navi.

Scelto l’avversario, inizia la prima parte della partita in cui il giocatore deve scegliere come posizionare le navi disponibili sul campo di battaglia, disegnato usando un TableLayout di ImageView.

Ogni casella di gioco, infatti, è descritta nel sistema usando una struttura Casella costituita dall’ImageView della griglia e un valore Boolean per settarla effettivamente occupata da una nave. Casella viene poi utilizzata in tutte le altre Activities attraverso la struttura CasellaPosition, che tiene traccia del nome dell’immagine usata su quella Casella.

Tutti i Motion events dell’activity SetupShipTableActivity sono intercettati dallo stesso metodo onTouch che gestisce sia le ImageView del campo di battaglia, che le ImageView delle barche.

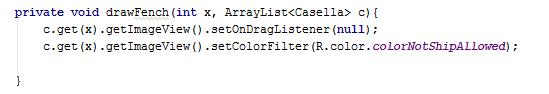
Cliccando una delle navi disponibili, il metodo onTouch registra l’evento MotionEvent.ACTION\_DOWN costruendo una Shadow dell’imageView attraverso la classe DragShadowBuilder, mentre rende invisibile quella cliccata dall’utente, dando l’idea di aver “staccato” l’immagine e di poterla trascinare fino al TableLayout.



Quando l’utente rilascia l’immagine, viene prodotto un MotionEvent.ACTION\_UP, che attiva il metodo onDrag().

Nel case DragEvent.ACTION\_DROP si gestisce singolarmente il cambiamento del Background delle caselle sulla griglia interessate dal Drop, in base alla grandezza della nave. La classe ShipPosition aggiorna i valori dell’ArrayList di Caselle e dell’ArrayList di Caselle Position, che rappresentano il campo di battaglia.



Le regole della “Battaglia Navale” non permettono di posizionare due navi a meno di una casella di distanza l’una dall’altra: se è stato possibile posizionare una nave in quelle caselle, si disegna un “recinto” intorno. 

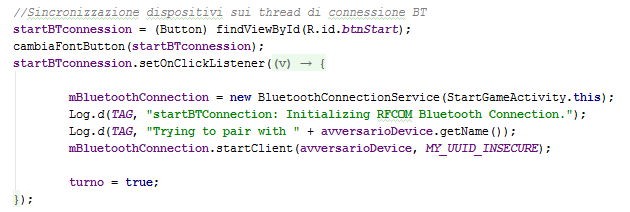
La classe ShipFence seleziona le singole ImageView intorno a quelle usate per posizionare la nave e, dopo aver invalidato il DragListener, applica un filtro sull’immagine presente, rendendo visibile all’utente la differenza tra queste e le caselle ancora disponibili ad ospitare navi.

* La partita

Nell’activity di gioco StartGameActivity si svolge la partita. È composta dal proprio campo di battaglia con le navi posizionate nell’activity precedente a sinistra e da quello dell’avversario a destra, entrambi TableLayouts.

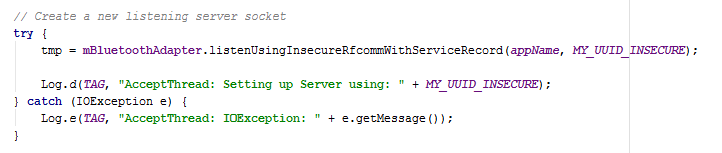
Quello a sinistra si aggiorna automaticamente durante il turno dell’avversario, mostrando al giocatore la mossa fatta; quello a destra ha ImageView cliccabili.

La connessione Bluetooth viene attivata in contemporanea dai due giocatori cliccando sul bottone Start, istanziando la classe BluetoothConnectionService e invocando il metodo startClient().

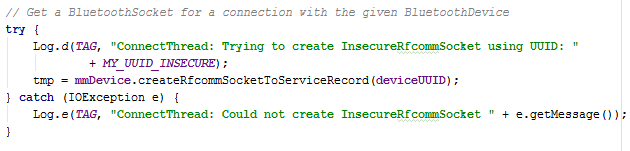


Lo scambio di dati avviene per mezzo di tre Threads:

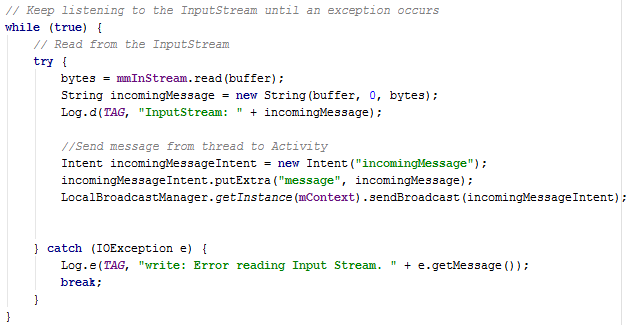
* AcceptThread(), che attende richieste di connessione in entrata: attraverso il metodo listenUsingInsecureRfcommWithServiceRecord() crea una BluetoothServerSocket sicura **in ascolto** usando l’indirizzo MAC del Device dell’avversario e il codice identificativo UUID. Termina solo quando la connessione stabilita è stata accettata da entrambi i dispositivi.



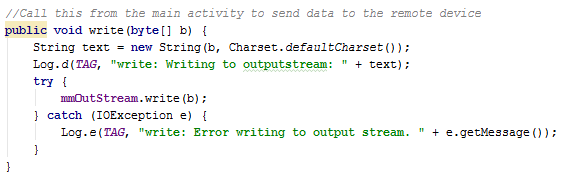
* ConnectThread(), che tenta di creare una connessione in uscita verso un altro dispositivo: attraverso il metodo createRfcommSocketToServiceRecord() crea una BluetoothSocket utilizzando il codice UUID. Terminando con successo, restituisce un canale di connessione tra i due giocatori.



* ConnectedThread(), che rimane **sempre** in ascolto fino all’invio o la ricezione di dati. Attraverso un InputStream riceve informazioni in entrata, mentre invia dati utilizzando un OutputStream: una volta letti i bytes arrivati, il Thread passa il messaggio ricevuto alla StartGameActivity attraverso un LocalBroadcastManager, sotto forma di Intent.



Il medoto write() di questo Thread, richiamato nell’Activity, permette lo scambio dei dati relativi alla partita, scrivendo sull’OutputStream.



Quando una delle ImageView del TableLayout viene cliccata, il ConnectedThread del dispositivo A scrive sul proprio OutputStream la posizione della casella cliccata, mentre il ConnectedThread del dispositivo B la riceve nell’InputStream. A questo punto si attiva il BroadcastReceiver del dispositivo B, che disegna l’evento nella casella corrispondente nella TableLayout di sinistra (un’esplosione se è stata colpita una nave, o dell’acqua altrimenti) e invia all’avversario la stessa informazione, scrivendo sull’OutputStream del proprio ConnectedThread.

Dopo aver disegnato su entrambi i dispositivi le relative immagini, i due ConnectedThread si rimettono in stato di attesa e l’ImageView diventa non più cliccabile attraverso invalidate().

I giocatori capiscono quando si tratta del proprio turno in base al colore dello sfondo dell’Activity, rosso nel caso di attesa e nero quando si può giocare.

Dopo che sono state colpite tutte le caselle della nave, che variano in base alla sua grandezza, compare l’immagine completa della nave.

Il gioco termina quando tutte le navi avversarie sono state scoperte: il dispositivo del giocatore che tra i due azzera per primo i contatori corrispondenti, invia l’ultimo messaggio via Bluetooth che notifica la fine della partita con un Dialog di sconfitta, e mostra un Dialog di vittoria. Entrambi i giocatori vengono ricondotti alla pagina iniziale del gioco, per poter iniziare una nuova partita.