

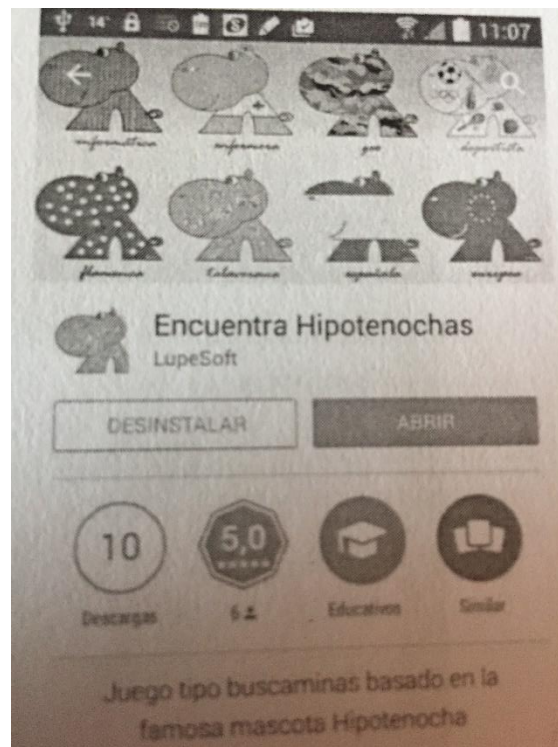


PRÁCTICA 2: ENCUENTRA HIPOTENOCHAS

El objetivo de esta tarea es programar un videojuego muy sencillo llamado encuentra hipotenochas. Este juego lo podéis descargar de google playstore para probarlo y examinar y analizar su funcionamiento.

Hipotenocha es la mascota de un proyecto de matemáticas de varios institutos de Castilla La Mancha. Nuestro objetivo es crear una App para que los chicos que participan en este proyecto puedan jugar con la mascota. En realidad el juego no es ninguna invención particular, es una copia del popular juego “Buscaminas”, solo que en lugar de buscar minas se buscan “Hipotenochas”.

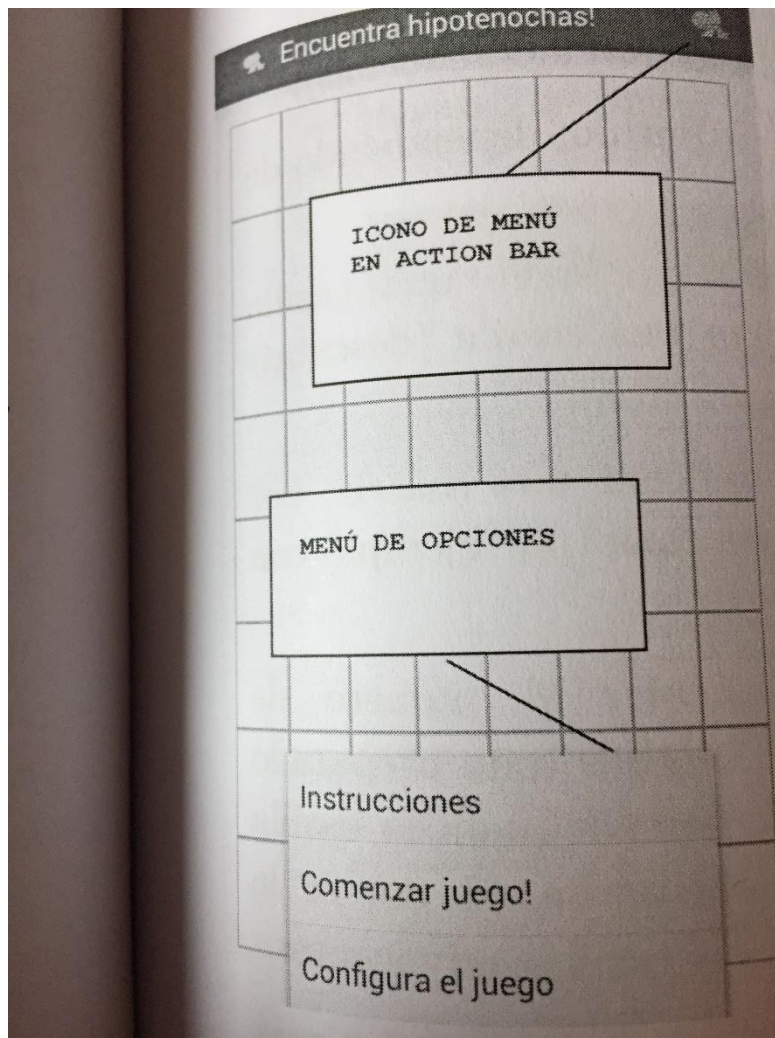
Vuestro objetivo es replicar y programar el mismo juego, dotándole si quierdes, de tu toque personal. Es decir, en lugar de buscar hipotenochas, puedes buscar algún otro tipo de personaje u objeto.



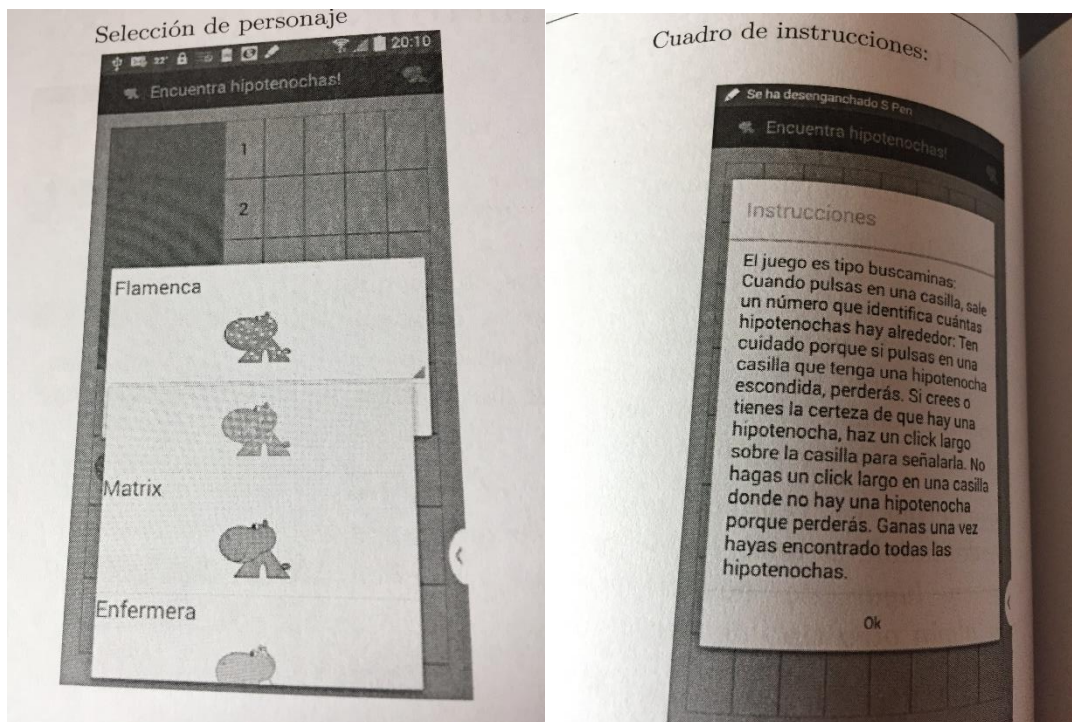
CONSEJO: Para que os sea más fácil la programación estudiar en profundidad cómo funciona el buscaminas. Tenéis muchísimos ejemplos en internet.

Lo que se os pide para vuestra App son los siguientes requisitos:

- Debe contener un menú en la ActionBar con las siguientes opciones:
 - Instrucciones (Opción de Menú): Sacará un Diálogo con las instrucciones del juego.
 - Nuevo Juego (Opción de Menú): Comenzará un nuevo juego, rellenando el tablero de hipotenochas y presentando la pantalla en blanco, dispuesta para que el jugador comience a buscarlas.
 - Configura el juego (Opción de Menú): Mostrará un Diálogo con RadioButton para poder seleccionar entre varios niveles de dificultad (Principiante, Amateur, Avanzado).
 - Selecciona personaje (Opción de Menú, con icono visible en la ActionBar): Permitirá seleccionar el tipo de personaje a buscar a través de un Diálogo con un Spinner que te permita seleccionar el gráfico a utilizar.



Los Diálogos que lanzan los menús son los siguientes:



Funcionamiento del juego:

El juego presentará al iniciarse un tablero para nivel principiante (8 filas x 8 columnas) con 10 hipotenochas escondidas. Para construir el tablero podéis utilizar un Layout de tipo GridView. Este tablero se puede configurar también en modo amateur (12 filas x 12 columnas) con 30 hipotenochas escondidas, y modo avanzado (16 filas x 16 columnas) con 60 hipotenochas escondidas.

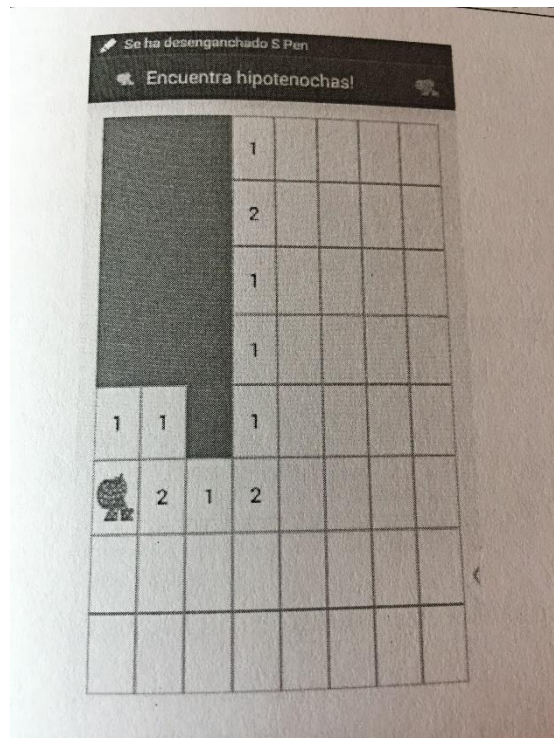
En cada celda del tablero habrá o bien un ImageButton o un Button, dependiendo de si hay hipotenocha o no.

El usuario podrá utilizar dos acciones: Clic o pulsación larga sobre una casilla (marcar posición de hipotenocha) y clic o pulsación corta sobre una casilla (descubrir contenido de una casilla).

Por tanto, el primer algoritmo que debéis escribir es para rellenar una matriz, por ejemplo de números enteros, donde tienes que ubicar 10 hipotenochas (por ejemplo con valor -1). -1 hay hipotenocha, >-1 no hay hipotenocha.

El segundo algoritmo que debéis programar es para calcular el número de hipotenochas que hay alrededor de cada casilla. De esta manera, tendréis todo preparado para que el usuario pueda ir descubriendo casillas. Si hay una hipotenocha en la casilla que el usuario ha pulsado con el dedo, se le mostrará la hipotenocha, si no, se le mostrará un número con el número de hipotenochas que tiene en las casillas adyacentes.

De forma opcional y para sacar más nota, podéis programar un algoritmo (os será muy fácil si lo hacéis recursivo) de manera que, si el usuario pulsa sobre una casilla donde hay 0 hipotenochoas alrededor, descubra automáticamente todas las casillas que hay alrededor (que seguro que no tienen hipotenochoas):

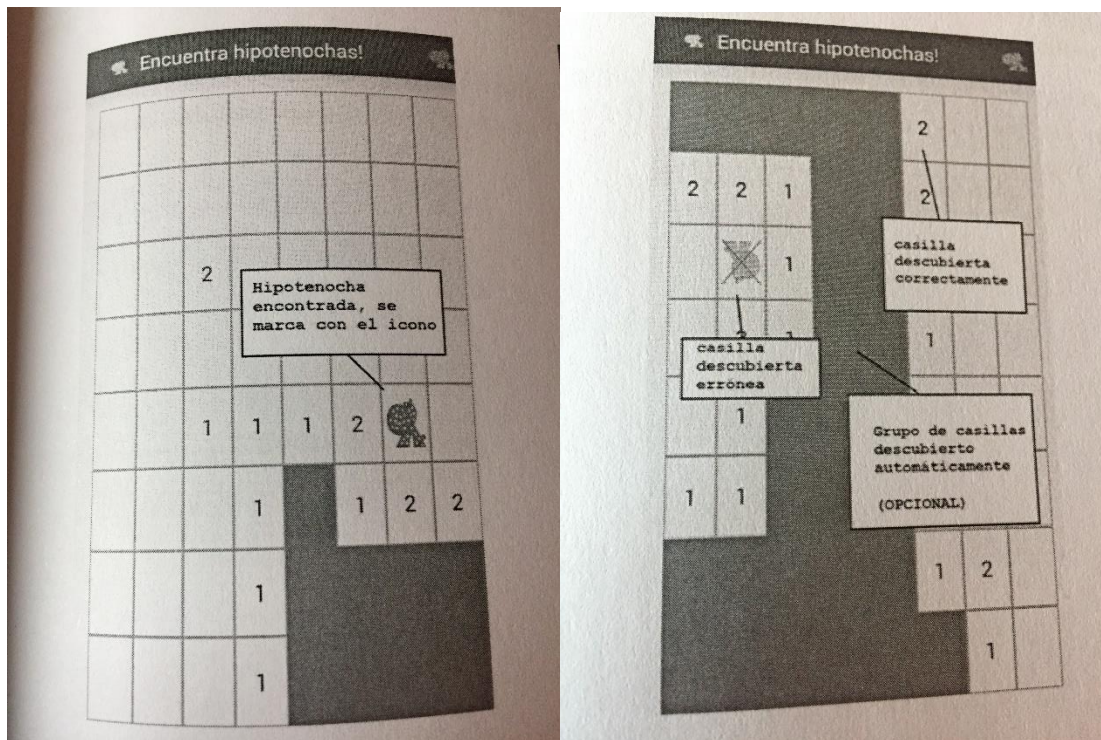


Desarrollo del juego:

El usuario puede efectuar dos acciones: Marcar hipotenocha (clic largo) o descubrir casilla (clic corto).

Tenemos cuatro situaciones distintas:

1. El usuario realiza un clic largo (onLongClick) sobre una casilla donde no hay una hipotenocha. El juego muestra la casilla descubierta y termina.
2. El usuario realiza un clic largo donde sí hay una hipotenocha, en este caso se marca y se indica que se ha encontrado una hipotenocha.
3. Se realiza un clic corto (onClick) en una casilla donde sí hay una hipotenocha. El juego termina con derrota mostrando una hipotenocha muerta (boca abajo y tachada).
4. El usuario realiza un clic corto (onClick) en una casilla donde no hay una hipotenocha. Se descubre el número que oculta.



Condición de victoria:

El usuario gana cuando ha ocurrido la situación 2 y ha marcado correctamente todas las hipotenochas.

Condición de derrota:

El usuario pierde cuando ha ocurrido la situación 1 o 3.

Criterios de evaluación:

- La aplicación contiene el menú de Opciones pedido en el enunciado (**1 punto**).
- La aplicación rellena el tablero correctamente con hipotenchas de forma aleatoria (**1 punto**).
- La aplicación calcula por cada casilla, cuántas hipotenchas hay alrededor (**2 puntos**).
- La aplicación tiene programado el evento onLongClick en cada casilla del tablero y responde de forma correcta ante las posibles posibilidades (fin de juego, hipotenocha encontrada) (**2 puntos**).
- La aplicación tiene programado el evento onClick en todas las casillas del tablero y responde de forma correcta ante las posibilidades (fin de juego, casilla descubierta) (**2 puntos**).
- La aplicación descubre automáticamente las casillas adyacentes cuando encuentra una casilla con valor 0 (ninguna hipotenocha en las casillas adyacentes) (**2 puntos**).