



ZÁPADOČESKÁ UNIVERZITA V PLZNI

MOBILNÍ KOMUNIKACE A ZAŘÍZENÍ

KIV/MKZ

Dokumentace semestrální práce

Vojtěch DANIŠÍK
A16B0019P
danisik@students.zcu.cz

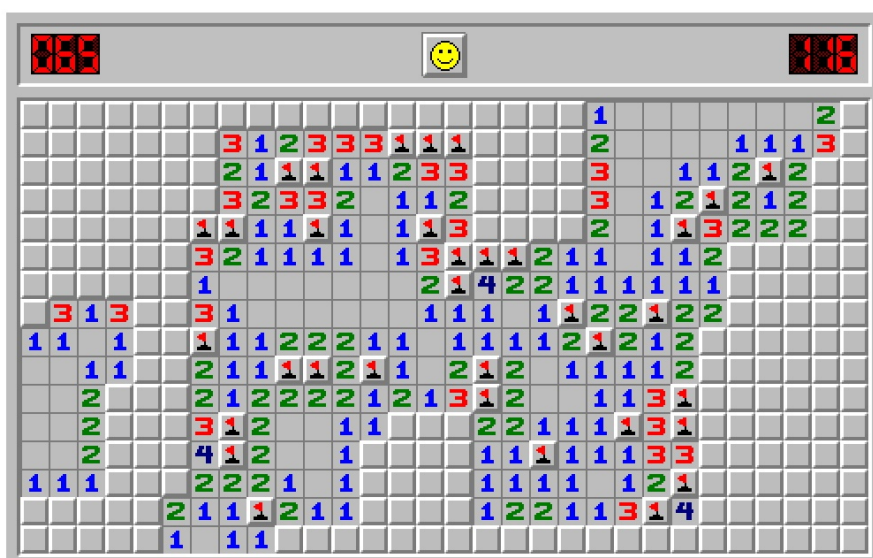
15. dubna 2019

Obsah

1	Zadání	2
2	Programátorská dokumentace	3
3	Uživatelská dokumentace	5
3.1	Vzhled aplikace	5
4	Řešené problémy	8
5	Testování	9
6	Závěr	10

1 Zadání

Jako zadání semestrální práce pro předmět MKZ byla vybrána hra **Hledání min - Minesweeper**. Hlavní podstatou hledání min je do stanoveného času označkovat pomocí vlajek políčka, která nebyla ještě odhalena a pod kterými se mohou nacházet bomby. Pokud hráč klikne na políčko, která obsahovala bombu, tak bomba vybuchne a hra končí neúspěchem. Počet vlajek ve hře je roven počtu bomb rozmístěných náhodně po celé hrací ploše. Pokud hráč klikne na políčko, které nesousedí ani s jednou bombou, je pole odkryto do té doby, dokud všechna nově objevená políčka nemají ve svém okolí alespoň jednu bombu. Čísla, která se ukáží po kliknutí na políčko značí počet bomb sousedících s aktuálně kliknutým políčkem. Vzhled této hry lze vidět na obrázku 1.



Obrázek 1: Hledání min - ukázka

2 Programátorská dokumentace

V programu bylo použito pro docílení správné funkcionality 7 tříd, z toho jedna abstraktní.

Třída **MainActivity** slouží jako hlavní třída aplikace. V této třídě se děje zobrazování již vytvořených struktur, jako jsou například časovač, hrací panel ale také i přiřazování aktivit pro tlačítka RESET a SET FLAG.

Třída **GameEngine** obsahuje funkční kód hrací plochy a tlačítek samotných. Při kliknutí na jednotlivé tlačítko je zavolána metoda *click*, která má za úkol zjistit mód ve kterém se uživatel nachází (zda odhaluje bomby nebo dává vlajky) a následně podle toho udělat určitou aktivitu. Zároveň se po každém kliknutí kontroluje, zda není hra dohrána pomocí metody *checkEnd*, ale i zda uživatel nezvolil tlačítko, které obsahuje bombu.

Třída **Generator** má za úkol vytvořit celou hrací plochu a všech tlačítek. Nejdříve jsou vytvořena políčka bez jakéhokoliv označení, pouze jsou náhodně vygenerovány pozice políček, na kterých bude vložena bomba. Po vygenerování všech políček probíhá vypočítání čísla políčka, které značí počet bomb, které se vyskytují v blízkosti aktuálně zpracovávaného políčka viz obrázek 2.



Obrázek 2: Sousední políčka, která mohou potencionálně obsahovat bombu

Abstraktní třída **BaseCell** má za úkol připravit strukturu pro políčko, které bude vloženo do hracího pole. Obsahuje proměnné udávající zda je políčko bomba, jestli je označené vlajkou, zda je odhledno a hodnotu udávající počet sousedních bomb doplněno o pozici v hracím poli.

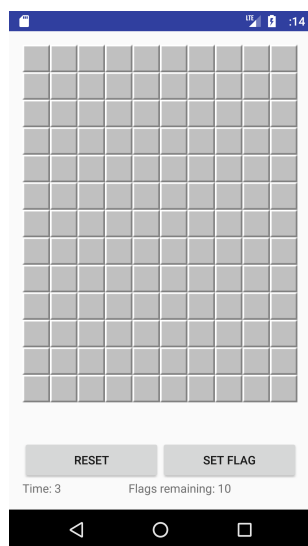
Třída **Cell** rozšiřuje námi vytvořenou abstraktní třídu **BaseCell**. Hlavním úkolem této třídy je přiřadit na základě stanovených hodnot (ve třídě **BaseCell**), jaký obrázek bude použit při vykreslení (při neodhalení, po odhalení zda se jedná o bombu nebo pouze obsahuje číslo udávající počet sousedních bomb).

Třída **Grid** zařizuje vytvoření veškerých struktur, které jsou dále posílány třídě MainActivity. Pro splnění tohoto účelu byla vytvořena private třída **GridAdapter**, která slouží jako most mezi námi vytvořenými daty a jejich vyobrazením.

3 Uživatelská dokumentace

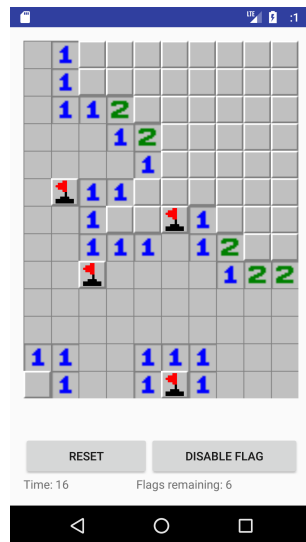
3.1 Vzhled aplikace

Výsledný vzhled aplikace lze vidět na několika obrázcích. První obrázek (viz 3) vyobrazuje aplikaci po jejím spuštění. Trochu se liší od původního vzhledu aplikace (viz 1), časovač a čítač vlajek je posunut pod hrací pole.



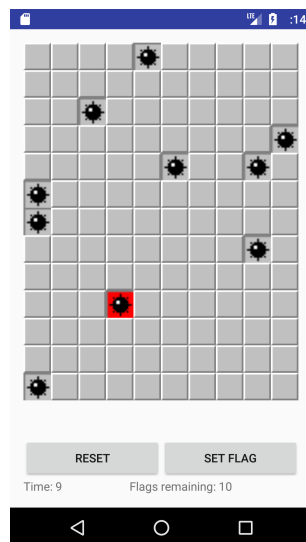
Obrázek 3: Vzhled aplikace

Do aplikace byly následně přidány i dvě tlačítka. Levé tlačítko *RESET* má za úkol vrátit již rozehranou hru do původního stavu (ať už uprostřed hry nebo po prohře). Druhé tlačítko *SET FLAG/DISABLE FLAG* má za úkol měnit stav hraní při kliknutí. Po prvním stisknutí je zapnut stav pokládání vlajek, zatímco po druhém je stav vrácen do původního a hráč může dále objevovat stále neobjevená políčka. Příklad vlajek lze vidět na obrázku 4.



Obrázek 4: Využití vlajek pro označení bomb

Pokud uživatel při odhalování hracího pole narazí na bombu, pak je tato bomba zvýrazněna, ostatní bomby jsou odhaleny a hra končí. Následně má uživatel možnost hru opakovat stisknutím tlačítka *RESET* nebo hru ukončit. Příklad kliknutí na bombu lze vidět na obrázku 5.



Obrázek 5: Bomba vybuchla - konec hry

Pokud uživatel hru dohraje, to znamená odhalí všechna pole neobsahující bombu a označí vlajkami všechna pole, která bombu obsahují, pak je

zobrazeno oznámení o dohrání hry a hra končí. Stejně jako u prohry může hráč začít hrát novou hru pomocí tlačítka *RESET* nebo hru ukončit. Příklad dohrání hry do úplného konce lze vidět na obrázku 6.



Obrázek 6: Odhalení všech políček a označení všech bomb - výhra

4 Řešené problémy

Při vyvíjení aplikace jsem narazil na jeden problém, který spočíval v nefungování třídy **Timer** a **TimerTask**. V aplikaci funguje časovač, který ukazuje uživatelům počet nahraných sekund v aktuální hře. Protože třídy **Timer** a **TimerTask** se často využívají v JAVA aplikacích, proto byly zvoleny i pro tuto aplikaci. Při implementování zmíněných tříd ale časovač nefungoval vůbec, důvody nebyly zjištěny. Proto jsem přestoupil na třídu **CountDownTimer**, která fungovala bez problémů.

5 Testování

Aplikace byla vyvíjena a testována v Android Studiu verze 3.3 (od firmy JetBrains) v jazyce JAVA. Při vývoji bylo využito JRE verze 1.8.0_152. Jako testovací mobil byl vybrán Nexus 5.2 s API verzí 24 a operačním systémem Android ve verzi 7.0 (Nougat).

Aplikace byla testována několika uživateli (převážně studenti ZČU), kteří při hraní hry nenarazili na žádný problém, který by měl fatální dopadek na běh aplikace.

6 Závěr

Vytvořená mobilní aplikace **Minesweeper**, neboli hledání min, splňuje základní funkcionalitu stejnojmenné hry. Uživatel je schopný hru bezproblémově dohrát a ukončit ji v jakémkoliv případě. Ovládání aplikace je velice jednoduché, stačí pouze klikat na políčka v hracím poli a podle zobrazeného čísla u některých políček odhadovat, na jakém políčku se můžou nacházet bomby. Bomby jsou rozmístěny náhodně, v každé hře jsou na jiné pozici.

Při testování se nevyskytly žádné chyby, které by měly zásadní vliv na funkčnost aplikace. Testovány byly především všechny stavy, které se mohli u této hry vyskytnout, jako například výhra/prohra, překročení limitu vlajek nebo označení políčka vlajkou i přes to, že už je odhalené.