

Navn: Roosbeh Moradi

Student: Morroo16

Student nr: 704279

Teknisk

Javaverson: 8

Emulator brukt: Nexus 5X API 24

Mine valg

Styrker

Detaljer som er lagt til :

Main page

Ved valg av spillernavn har jeg lagt til 'Ready ' knapper som lyser for å varsle andre spilleren om at en er klar og knappen blir låst samtidig som tekstfeltet ikke kan redigeres. Dette gir en fordel ved en fremtidig online løsning.

Spillerne kan ikke :

- hete det samme
- hete ' TTTbot'
- kan ikke klicke play om begge tekstfelt er tomme. Er den ene ready og andre ikke er mens tekstfelt likevel er fylt toastes en melding om at spillerne må klikke ready. Btn 'Letsplay' lyses opp og varsler om at det er mulig å starte spillet når alt er greit.
- Knapper lyser og varsler når en funksjon bør/kan taes i bruk.

InGame page

- Save btn disabled når et spill ikke er ferdig. Lyser rødt og varsler når det er mulig å save.
- Btn quitP1 og btnQuitP2 blir disabled når en av spillerne er TTTbot ellers bare en vanlig knapp hvis mulig å slutte å spille og dermed vil TTTbot ta over spillet.
- Btn Exit forlater, avslutter spillet og returnerer til hovedside og er tilgjengelig gjennom hele spillet.
- En av spillerne kan avslutte spillet under spillet. Deretter vil TTTbot ta over.

HighScore page

- Sorteres etter antall wins.
- Har 2 spillere like mange wins rangeres de etter raskest tid
- Gir melding om ny eller eksisterende vinner er lagt til ved 'save' klikk
- Har muligheten til å slette tabellen

Svakheter

Bruker mye ressurser fordi det er 3 Activities. Planen var å bruke kun en Activity med flere fragments, men ga opp etter ørten forskjellige måter å gjøre det på. Forvirret meg mye med hva som skulle implementeres hvor i fragment klassene og Activity klassene. Uansett hvordan så ble utseende helt annerledes enn designen var. Den jeg kjørte den ble widgetsene gjenværende i det fragmentbytte tok plass. Det endte opp med at jeg brukte en activity for hver side der hver er basert på flere fragments. Gikk for denne løsningen fordi det gjorde visse ting lettere for meg som f.eks ved plassering av widgets eller når størrelsen på hver justeres eller ved fremtidig endring av gui. Dette gjør at innhold i andre fragments ikke påvirkes når ting flyttes på noe som var et utrolig herk å styre med. Innså de siste dagene hvordan fragments egentlig skal fungere så jeg prøvde å løse den siste listView oppgaven med riktig bruk av fragments de siste 3 dagene , men fikk ikke tid da fragmentet også var avhengig av at jeg endret alt på SignInActivity også. Så en del eksempler på nettet med bruk av FrameLayout og LinearLayout for oppsett av spillknappene , men ville unngå kopiering så jeg prøvde meg på Relative i stedet. Årsaken til at en skjerm er delt opp i flere fragmenter er at resultatet ble alltid helt annerledes enn det design viewet i android studio fremla og de endret seg under trykking både med Constraint layout og Linear. Ved å dele de opp i flere fragments gjorde det lettere for meg å tilpasse størrelsene uten å påvirke andre widgets. Klar over at dette er en veldig feil måte å bruke fragments på , men jeg misforstod som sagt bruken og prøvde å rette på det og samle alle widgets på en fragment i ettertid og bruke FragmentManager istedenfor Intent osv, men rakk ikke og håper ikke dette trekker så alt for mye .

Andre svakheter er kode struktur. Fordi jeg var veldig usikker på sammenhengene mellom kodene i Android, hva som er mulig og ikke mulig, bestemte jeg meg for å føre inn meste parten av knappefunksjoner og slikt i samme klasse for å sikre meg at det ikke var problemet til at ting ikke fungerte for så å rydde opp og fordele til flere klasser til slutt. Men det endte dessverre uheldigvis opp med at jeg ikke rakk å gjøre noe mer enn som så i InGameActivity.

Googleplay deployment

Fikk dessverre ikke til å laste opp til playstore da jeg hadde en hel haug av problemer med godkjenning av googleplay invitasjon og buildfail meldinger ved pushing til git .

Slides på itsLearning ga ingen konkret forklaring på at mail sendes til skoleadresse og hvordan en går frem med disse feilmedlingene.

Invitation is not available

This invitation is either invalid, expired, or already used by a different account. You can contact a developer account admin and request for the invitation to be sent again.

Mitt problem var at jeg ikke fikk funnet ut av dette før siste liten og da var det allerede for sent å styre med nytt konto på git med skolemailen.

Klasser og Metoders funksjoner

MainPageActivity

'Klassen er hovedsiden og tar imot brukernavn og overfører til IngameActivity / HighScoreList'

onCreate(): Utfører handlingene angitt ved oppstart

findViewById(): Finner widgets i View

onClick(): Angir knappeunksjonene

disableEdtAndChangeBtnColor(EditText edtP1, Button btnReady1, Button btnLetsPlay):

avkobler endringsmuligheter i EditText feltet og endrer knappefarger.

goToHighScorePage(): Bytter siden til HighScorePage med Intent

startGame(): Sender over brukernes valgte navn og bytter over til siden InGameActivity

getPlayerName(EdtText): tar imot en 'Edittext' og returnerer navnet som er valgt

HighScoreActivity

'Klassen viser en liste over beste spillere rangert etter antall vunnet og fullført på best tid'

onCreate(): Utfører handlingene angitt ved oppstart

findViewById(): Finner widgets i View

onClick(): Angir knappeunksjonene

updateListView(namelist): tar imot ListViewets ArrayList, sorterer og Oppdaterer ListViewet med en adabter.

makeListView(List<WinnerData> dataList): tar imot en navneliste som lagrer data fra databasen og oppretter ListViewet

InGameActivity

'Klassen styrer brettspillet, Kommuniserer med Firebase skylagring og telefonens interne lagring'

initPlayers(): tar imot navn fra Mainpage og føyer det til P1 og eller P2 texview, 'TTTbot' som default hvis mottak er tomt.

newMatch(): starts a new match

humanPlayerSetButtonXorO(Button): Setter player1 som 'X' og player2 som 'O'

setXorOLayout(Button btn, String XorO): Angir 'X' fargen Rød og 'O' fargen Sort og størrelsen

disableQuitBtns(): avkobler 'btnQuitP1 og P2'

disableBtnSave() avkobler 'save' button

enableBtnSave() tilgjengeliggjør 'save' button

setRandomOnClick() samler opp ledige knapper og setter tilfeldig knapp til 'X' eller 'O' hvis spiller er 'bot'

switchPlayerToBot(String player, String PlayerTurn, String randomXorO) : Bytter om P1 eller P2 til bot og avgjør om bot skal være 'X' eller 'O'

setWinner(String Player): Angir vinneren

matchController(): Dømmer kampen /avgjør vinneren eller uavgjort

drawGame(): avgjør uavgjort

isButtonXorO(Button btn, String XorX): sjekker om knappen er X eller O

setPlayerTurn(String player): setter hvem sin tur det er

enableGameButtons(boolean trueOrFalse): Tilgjengeliggjør spillknappene

resetGameButtons(): Tilbakestiller spillknappene

addButtonsToList(): Samler spillknappene i en ArrayList

resetTimer() Tilbakestiller tiden

DBhandler

DBsoure.class

'Håndterer Crud operasjoner'

DBsoure(Context context): tar imot en context og oppretter en et SQLhelper objekt

open(): Åpner forbindelse med Databasen for lesing og skrivning til database

close(): lukker forbindelsen

addNewWinner(String winner, String time, int countWin) : Legger til en ny winner

getAllWinnerData(): henter all data om winnere fra tabellen Highcorelist

nameExists(String name): sjekker om bruker eksisterer

incrementWins(String name, int wins): øker antall wins med 1

clearTableHighscorelist(): sletter tabellen highscorelist

WinnerData.class

'Placeholder for henting og innsetting av data fra og til databasen med Cursor'

setCountWin(): setter antall wins i Cursor

getCountWin: Henter

getName() Henter navn (ikke i bruk)

setName(String name): setter navn i Cursor
getTime(): henter tidenspillet ble fullført på (ikke i bruk)
setTime(String time): setter tiden i Cursor
getId(): henter id (ikke i bruk)
setId(long id): set id i Cursor

Messenger.class

showMessageDialog(Context contex, String title, String msg): varsler bruker med en tittel og avsluttes ved berøring.

toastMessage(Context context, String message) : varsler bruker med en kortvarig melding .

build.gradle

- *Project added :*

classpath 'com.google.gms:google-services:3.2.0'

- *Module:app lagt til dependencies :*

compile 'com.google.firebase:firebase-core:15.0.0'

compile 'com.google.firebase:firebase-database:15.0.0'

compile 'com.google.firebase:firebase-crash:15.0.0'

compile 'com.firebase:firebase-client-android:2.5.2'

plugin: 'com.google.gms.google-services'

Ressurser

Activities

MainPageActivity

- `<android: maxLine=1 >` i `sign_in_bottom_fragment.xml`:

(Mulig slike små detaljer er unødvendig ,men likevel)

<https://stackoverflow.com/questions/38936310/remove-next-line-visibility-in-multiline-edittext-android/38936455>

HighscoreActivity

ListView i Activityhighscore:

Eksperimenterte litt med ListView. Til å begynne med slet jeg litt med å få opp en ListView frem i det jeg addet items. Fant senere ut at jeg manglet å varsle ArrayAdapteren med `notifyDataSetChanged()` på slutten. Ressursen som bidro til å oppdage dette:

https://www.youtube.com/watch?v=ws_p8LJ4Uq8

ellers tittet på øvings kode eksempler

updateList()

Sortering av antall wins og tid med java8 ga natural order (ascending/synkende).

Brukte så ressurser fra denne nettsiden for å få det stigende:

<https://reversecoding.net/java-8-comparator-how-to-sort-a-list/>

InGameActivity

Valgte denne måten å endre color på btns på fordi det var lettere å resette knappen original igjen ved onClick.
Linjen lånt fra:

<https://stackoverflow.com/questions/14802354/how-to-reset-a-buttons-background-color-to-default>

```
btnRetry.setBackground().setColorFilter(Color.YELLOW,  
PorterDuff.Mode.MULTIPLY);  
btnExit.setBackground().setColorFilter(Color.YELLOW,  
PorterDuff.Mode.MULTIPLY);  
btnSave.setBackground().setColorFilter(Color.RED,  
PorterDuff.Mode.MULTIPLY);
```

- `setwinner()`

```
variabelen long millSec = SystemClock.elapsedRealtime() - timer.getBase();
```

<https://stackoverflow.com/questions/526524/android-get-time-of-chronometer-widget>
Erstattet long med double og deretter delte jeg verdien på 1000 og fikk seknder.

- `resetTimer();`

```
timer.setBase(SystemClock.elapsedRealtime());
```

<https://stackoverflow.com/questions/32651129/how-to-start-chronometer-from-0000>

- `toastMessage():`

<https://www.mkyong.com/android/android-toast-example/>

Database

SQLhelper

Var ikke så mange andre måter å gjøre det på enn som eksempel kodene

DBsource

`incrementWins()` SQL Spørringen funnet hos:

https://www.tutorialspoint.com/sqlite/sqlite_update_query.htm (eksempel under 'syntax')

`clearDatabase()` ressurser hentet fra Androids API

<https://developer.android.com/training/data-storage/sqlite.html#java>

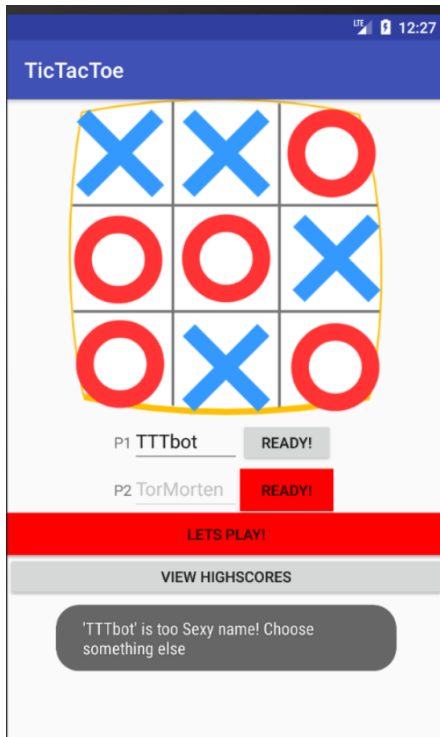
Ellers noe fra eksempel kodene til forelesningen

Firestore (i InGameActivity)

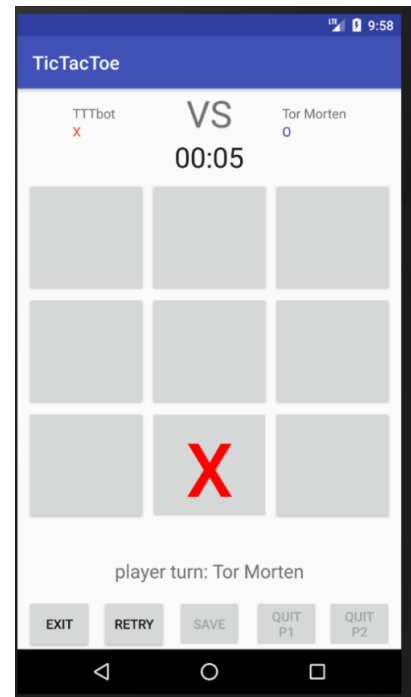
<https://developer.android.com/studio/write/firebase> installasjon

<https://firebase.google.com/docs/android/setup?authuser=0> bruker veiledning og mer installasjon

Screenshots

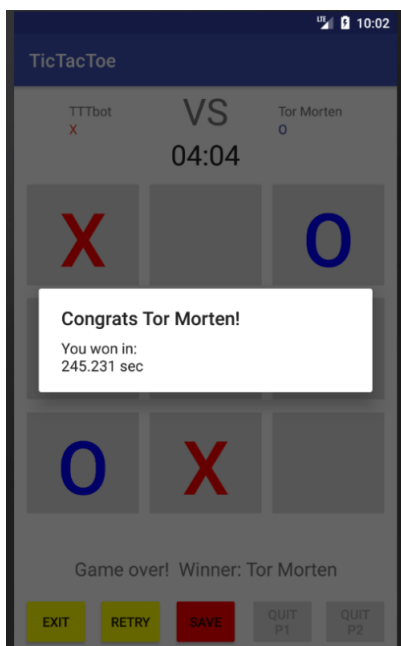


MainPage (locked if TTTbot)



InGame

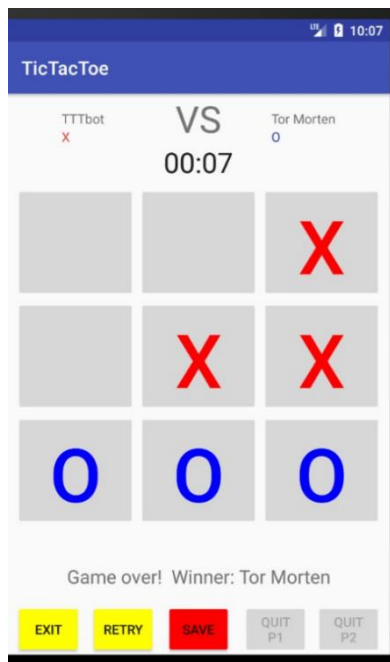
klikk 'Lets Play' →



Win Dialog

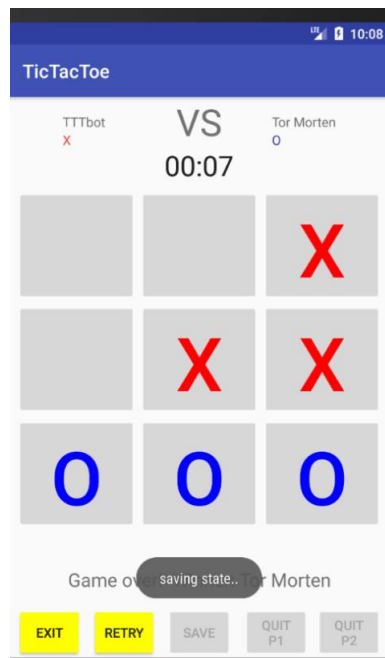


lose



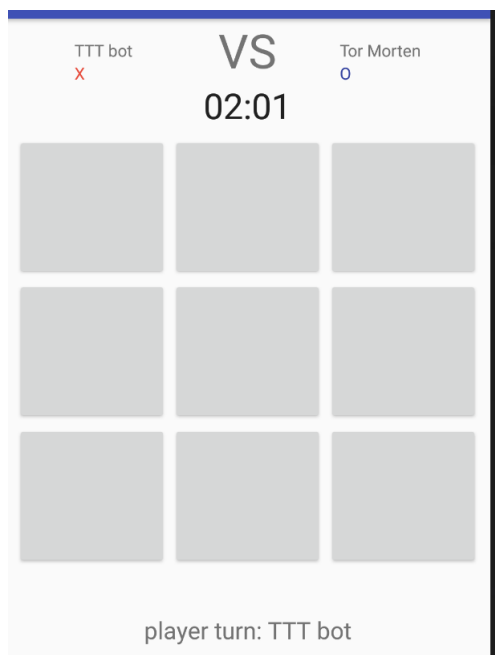
Menu if finished

klick save →



toast message

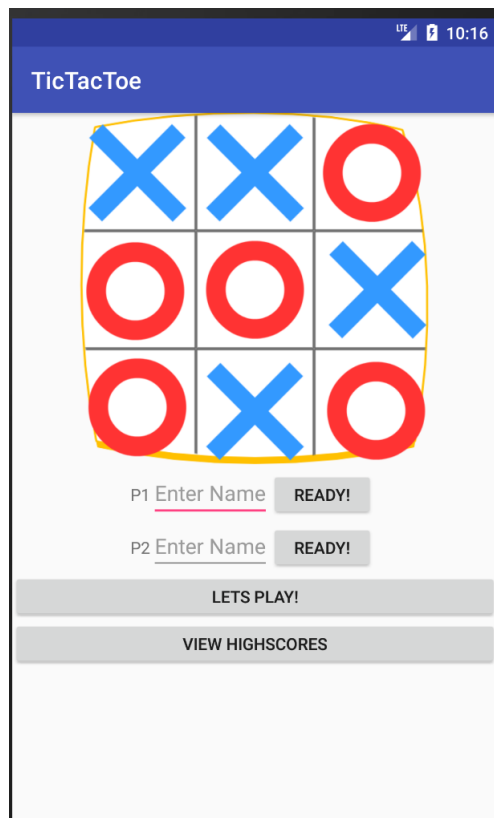
retry →



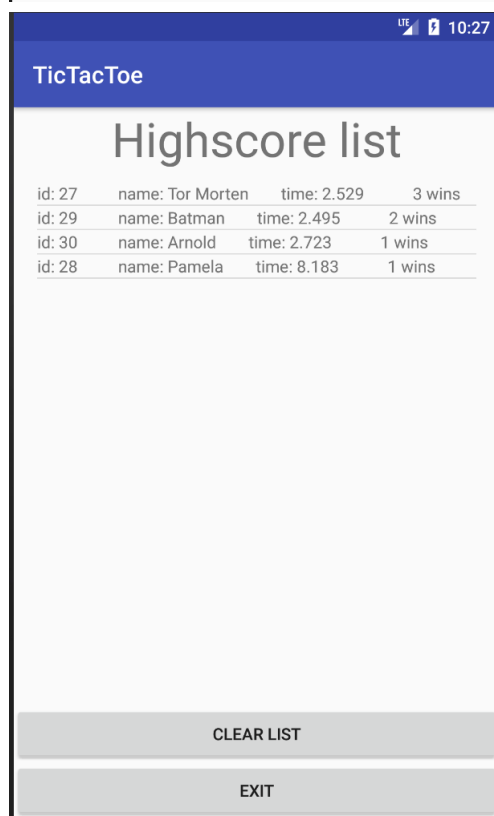
reset



Versus mode klick Quit button

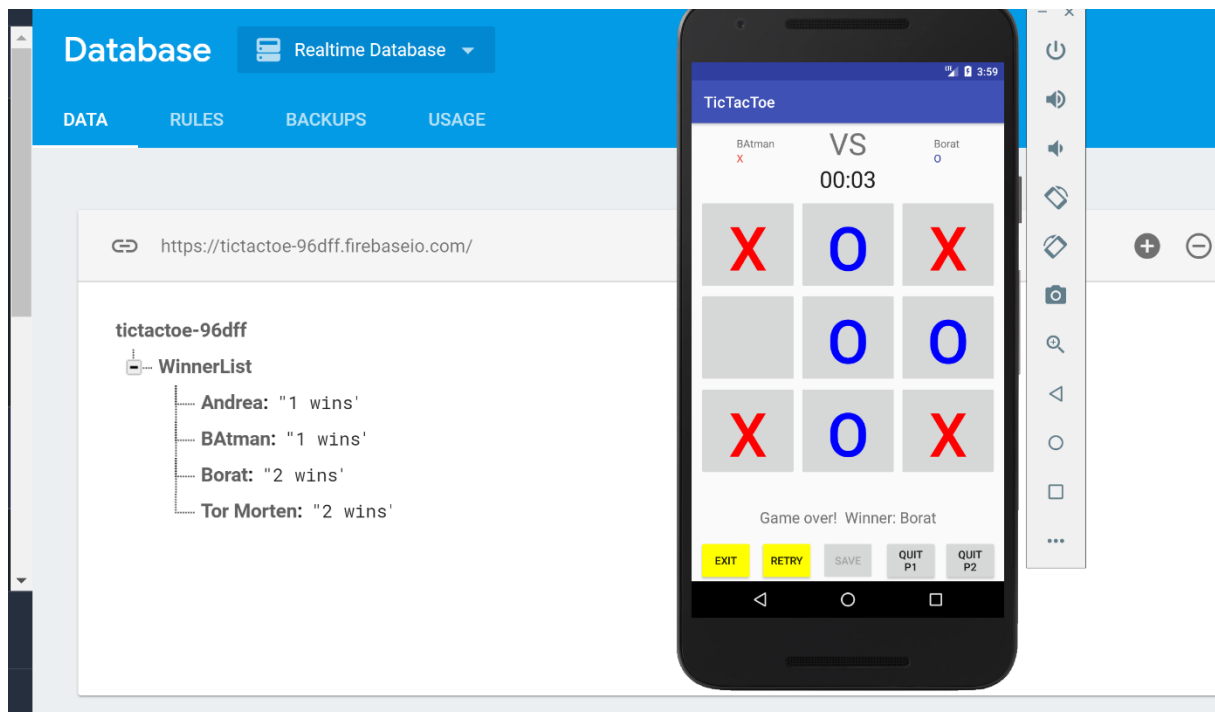
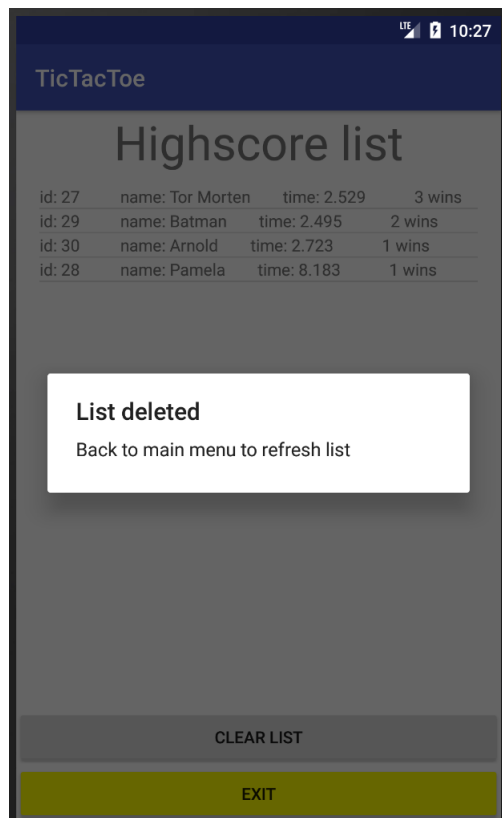


exit→



Klikk 'View Highscores'

Klikk 'Clear list' →



Firestore

[illegible]