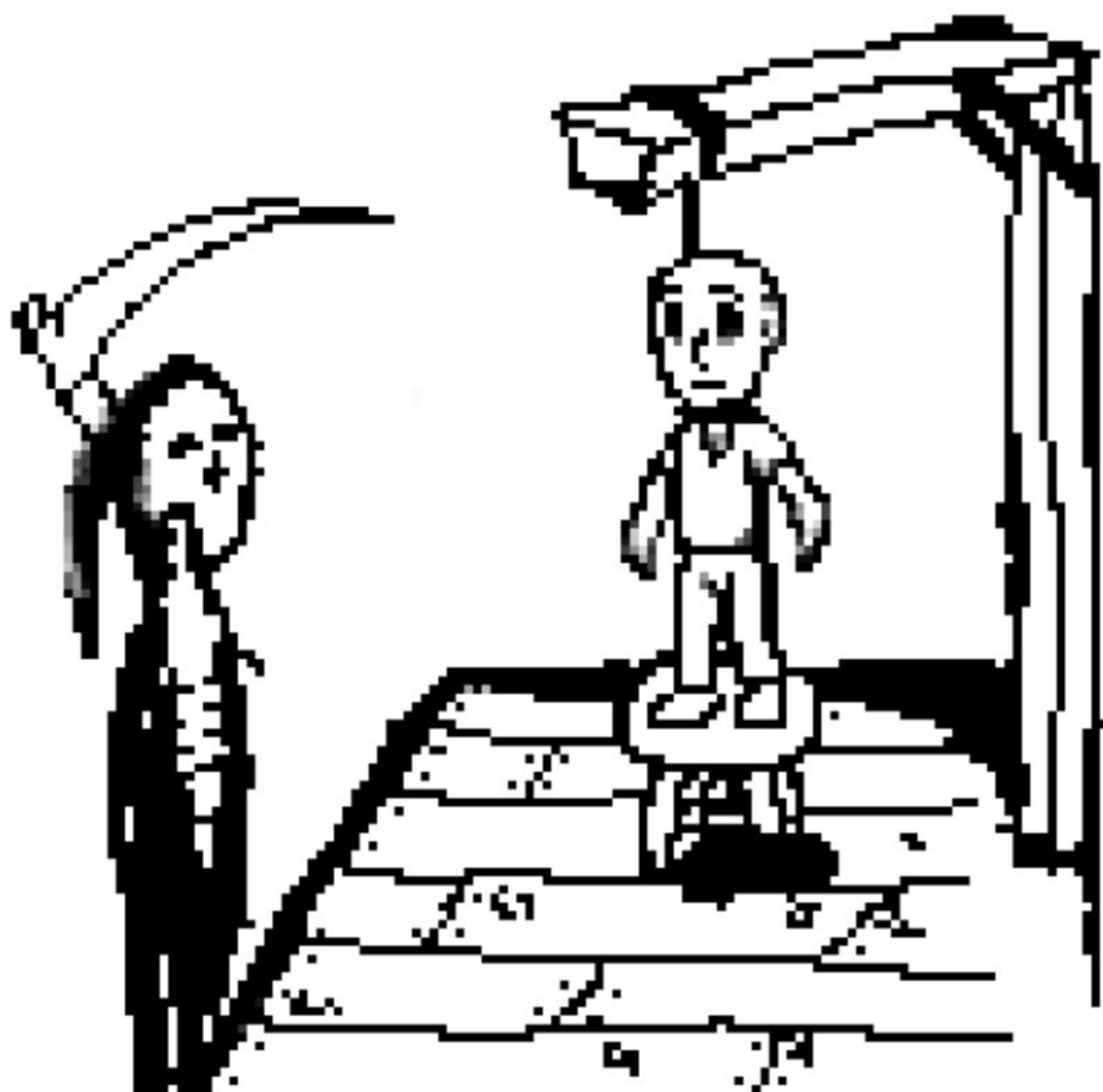


# DOKUMENTASJON HANGMAN

En "Hangman" android app, utviklet av: Anders Simonsen, s198739

# HANGMAN



Dette er dokumentasjonen til en variant av det kjente spillet hangman.

Spillet er utviklet som en oppgave i faget Applikasjonsutvikling på HiOA.

## **FORORD**

Dette er dokumentasjonen for mappeoppgave 1 i faget applikasjonsutvikling ved Høgskolen i Oslo og Akershus høsten 2014.

Dokumentet legger fram bakgrunn for produkt, rammebetingelser, verktøy og tilleggsfunksjonalitet.

Dokumentasjonen er beregnet for faglærer og sensor.

### **Litt om det tekniske bak appen og testing:**

Minimum API er satt til 15, og target API til 22, jeg har ikke testet på alle API'ene imellom.

Jeg har kjørt tester på emulatoren med API 19,

Jeg har også testet litt designet på min gamle Samsung Galaxy s2 som kjører API 17.

## OPPGAVEBESKRIVELSE

Det skal lages et Hangman-spill, for android platformen. Brukeren skal kunne velge om han/hun vil se regler, starte spill, endre språk eller avslutte applikasjonen. Brukeren skal kunne gjette bokstaver som til sammen former et ord.

## RAMMEBETINGELSER

- Android platformen
- XML

## UTVIKLERVERKTØY

Android Studio IDE

## KRAV

- Man skal kunne velge om man vil se regler, starte spill, endre språk, eller avslutte applikasjonen.
- Man skal kunne registrere hvilke bokstaver som er gjettet.
- Riktige bokstaver settes på riktig plass i ordet og gale bokstaver vises et annet sted på skjermen.
- For hvert galt svar endres galgebildet til man er hengt.
- Spillet skal kunne tilpasses Norsk og Engelsk språk.
- Registrere og vise hvor mange spill man har vunnet og hvor mange har tapt.

## FUNKSJONALITET

Spillet er kodet i Android Studio. Defaultspråk er satt til engelsk, men det er også lagt til norsk språk.

Jeg har valgt å legge til litt mere funksjonalitet enn i kravene. Istedenfor å registrere og vise hvor mange spill som er vunnet og tapt, har jeg gjort det slik at du sanker poeng for hvert riktig ord man har gjettet på rad. Poengsummen blir nullstilt når du taper. Jeg har også laget en highscoreliste som holder opptil 10 highscores, og implementert denne listen i *SQLITE*. Dersom du ved "Game Over" har flere poeng enn den største highscoren i lista så blir det opprettet et highscore objekt med navn, rundenr og poengsum. som blir lagt til i en *SQLITE* database. Highscorelisten kan også vises fra MainActivity metoden, via knappen "Highscore".

## DESIGN

Spillet er designet i Android Studio IDE. Jeg har valgt å kjøre litt enkel pixel-art til å prege appen.

Alle knappene har to til tre tilstander:

uttrykket, nøytral	trykket, riktig bokstav	trykket, feil bokstav
		

Kombinasjonene av svart, hvitt og grønt går igjen:

Logo	Bakgrunn	Spillgrafikk
		

I logoen har jeg prøvd å følge stilguiden til google:

<https://www.google.com/design/spec/style/icons.html#>

Den er designet i vectorgrafikk programmet inkscape.

Resten av grafikken, knapper etc. er laget i tegneprogrammet GIMP.

## Android designprinsipper

Simplify my life:

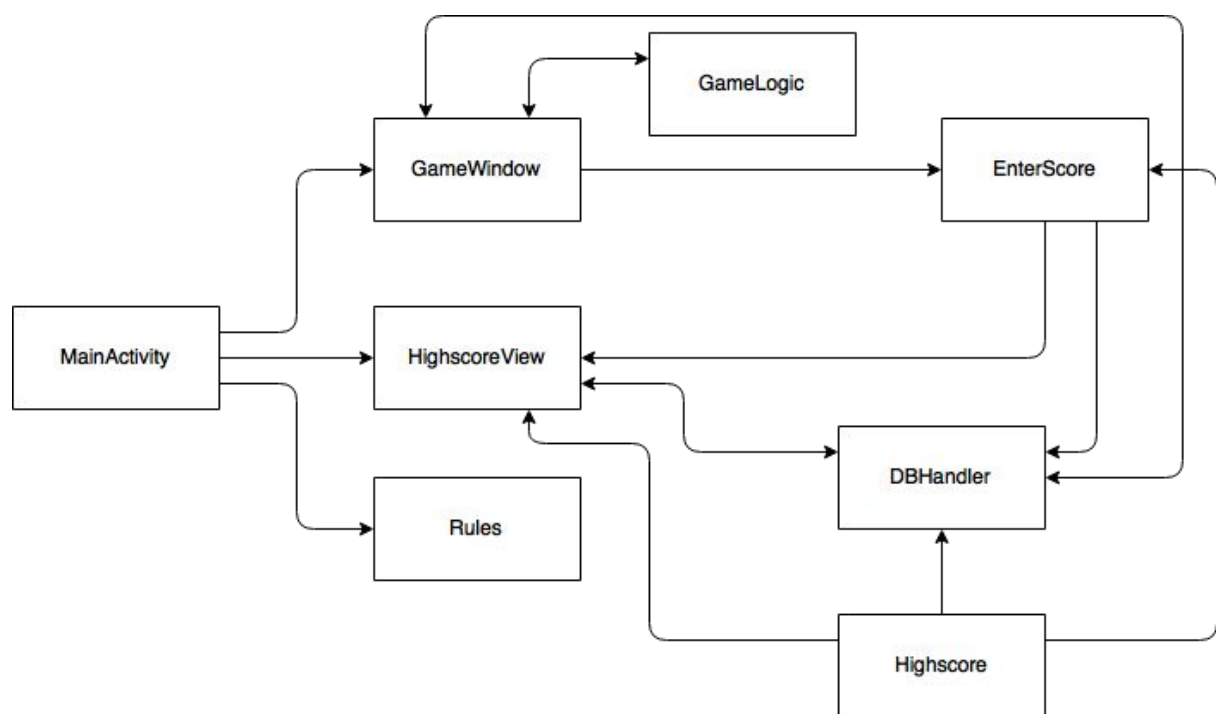
- Keep it brief: Applikasjonen benytter seg av korte fraser og enkle ord.
- Only show what I need when I need it: Applikasjonen viser deg bare det du trenger når du trenger det.

Enchant me:

- Bruk av enkle rene layouter, uten for mye dill dall, og rene fonter.
- Prøver å forenkle og gjøre applikasjonen ren og vakker.

## KODE :

- Klassen **MainActivity.java** er hovedaktiviteten brukes som et menysystem og er det første skjermbildet man ser når man starter appen, den bruker Layouten *activity\_main.xml*
- Klassen **GameWindow.java** er en aktivitet som styrer spillet startes opp av *MainActivity.java* sin *newGame(View view)* metode; I dette vinduet styrer spillet, og inneholder mange metoder. Den kommuniserer med klassen *GameLogic.java* og bruker Layouten *activity\_gamewindow.xml*
- Klassen **GameLogic.java** er en vanlig javaklasse som inneholder en del logikk som er skilt ut av *GameWindow* klassen.
- Klassen **Highscore.java** er en vanlig javaklasse som definerer et Highscore objekt.
- Klassen **DBHandler.java** oppretter et databaseobjekt og inneholder metoder for å opprette og styre en SQLITE database av Highscore objekter.
- Klassen *EnterScore.java* startes av *GameWindow.java* sin *GameOver()* metode, dersom du har oppnådd ny highscore, og er en aktivitet som benytter seg av layouten *activity\_enterscore.xml* som tillater deg å Skrive inn navnet ditt for å komme i highscorelisten.
- Klassen **HighscoreView.java** er en aktivitet som viser highscorelisten. Den bruker layouten *activity\_highscoreview.xml*, denne aktiviteten startes enten av *MainActivity* sin *Highscores(View view)* metode, eller fra *GameView* sin *gameOver()* metode hvis *DBHandler* sin *isHighscore(int points)* metode returnerer true;
- Klassen **Rules.java** er en aktivitet som viser spillets regler i et *TextView*, benytter seg av layouten *activity\_rules.xml* for den grafiske visningen. aktiviteten startes fra *MainActivity* sin *rules(View view)* metode.

**ILLUSTRASJON AV KLASSENEES KOMMUNIKASJON SEG IMELLOM:**

Dette bildet viser klassene i appen, og hvordan de kommuniserer med hverandre.

## SKJERMBILDER AV DE ULIKE AKTIVITETENE

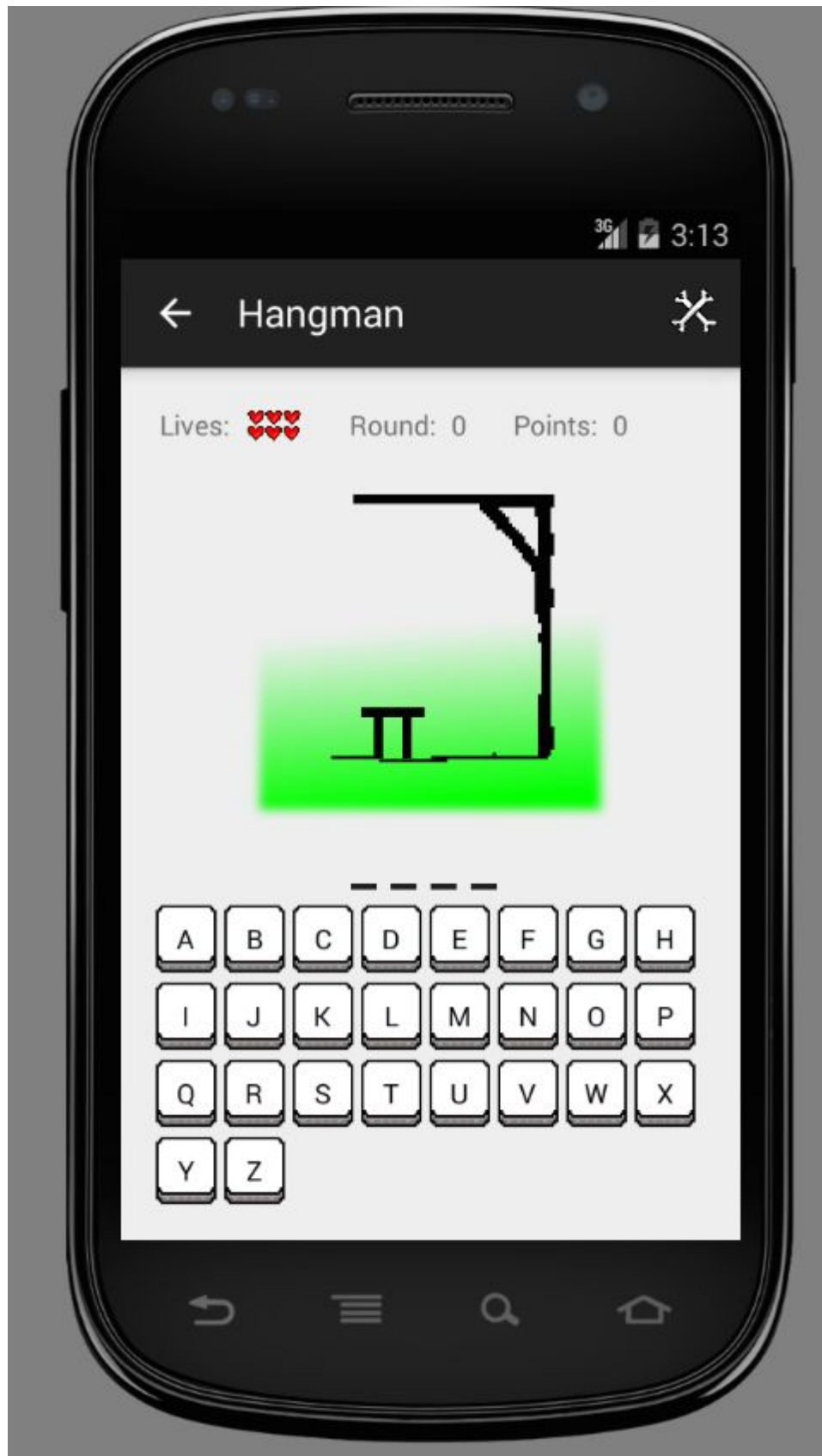


**MainActivity**, denne aktiviteten brukes som et menysystem. Med valg av språk med flaggknapper, og tre andre valg: "New game", "Highscore" og "Rules".

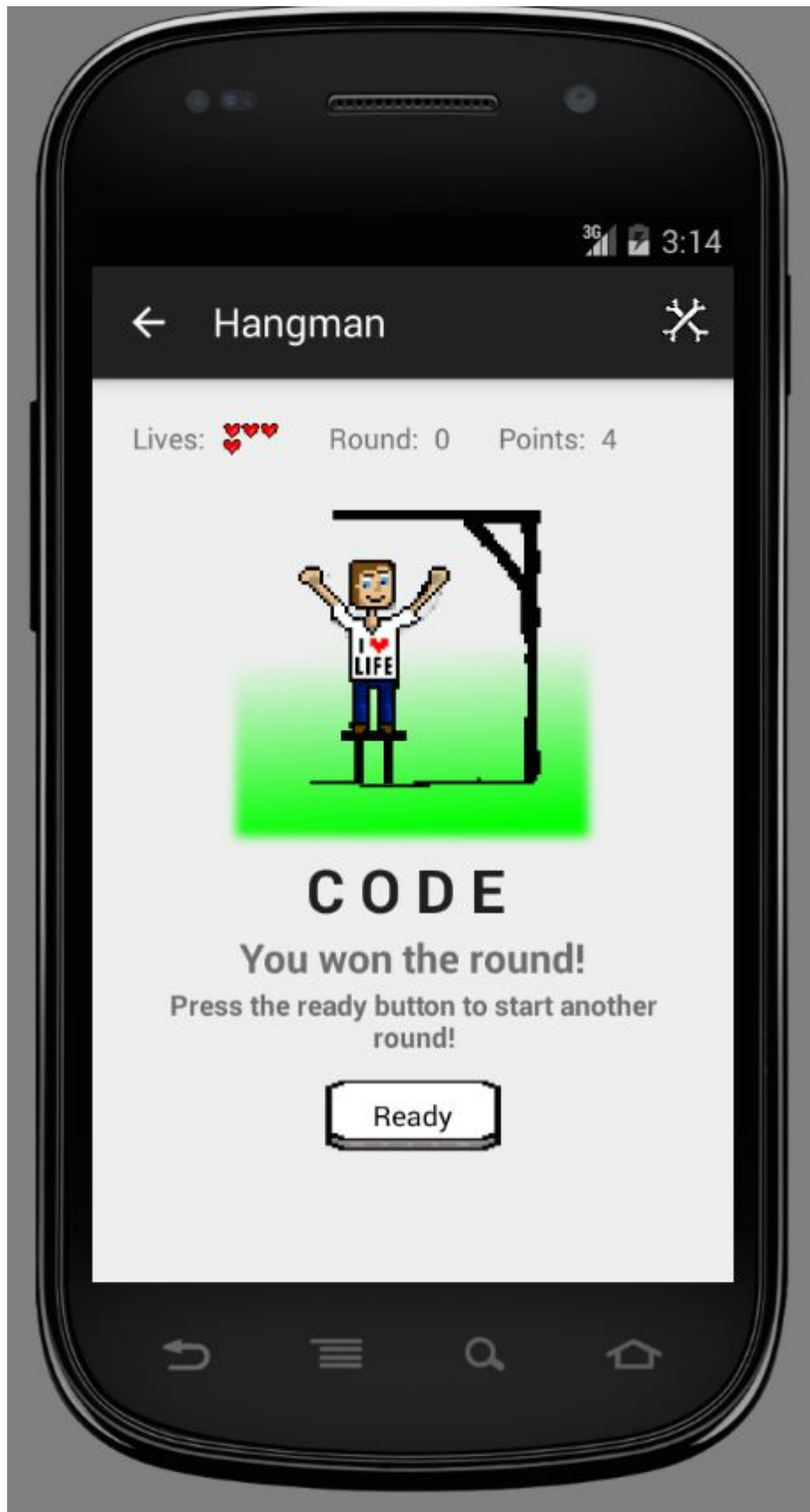
- **"NEW GAME"** knappen tar deg til GameWindow aktiviteten.
- **"HIGHSCORE"** knappen tar deg til HighscoreView aktiviteten

- **“RULES”** knappen tar deg til Rules aktiviteten





**GameWindow.** Viser et HUD(head up display), galgegrafikk, representasjon av ordet du skal gjette og et tastatur.



En variant av GameWindow. Når man har gjettest riktig ord og vunnet runden vises denne skjermen. Trykk på “ready” for å starte neste runde, hvor det genereres et nytt ord du må gjettest. Du fortsetter å øke på poengsummen din fra forrige runde.



Første gang du gjetter feil: (du blir satt på krakken under galgen)



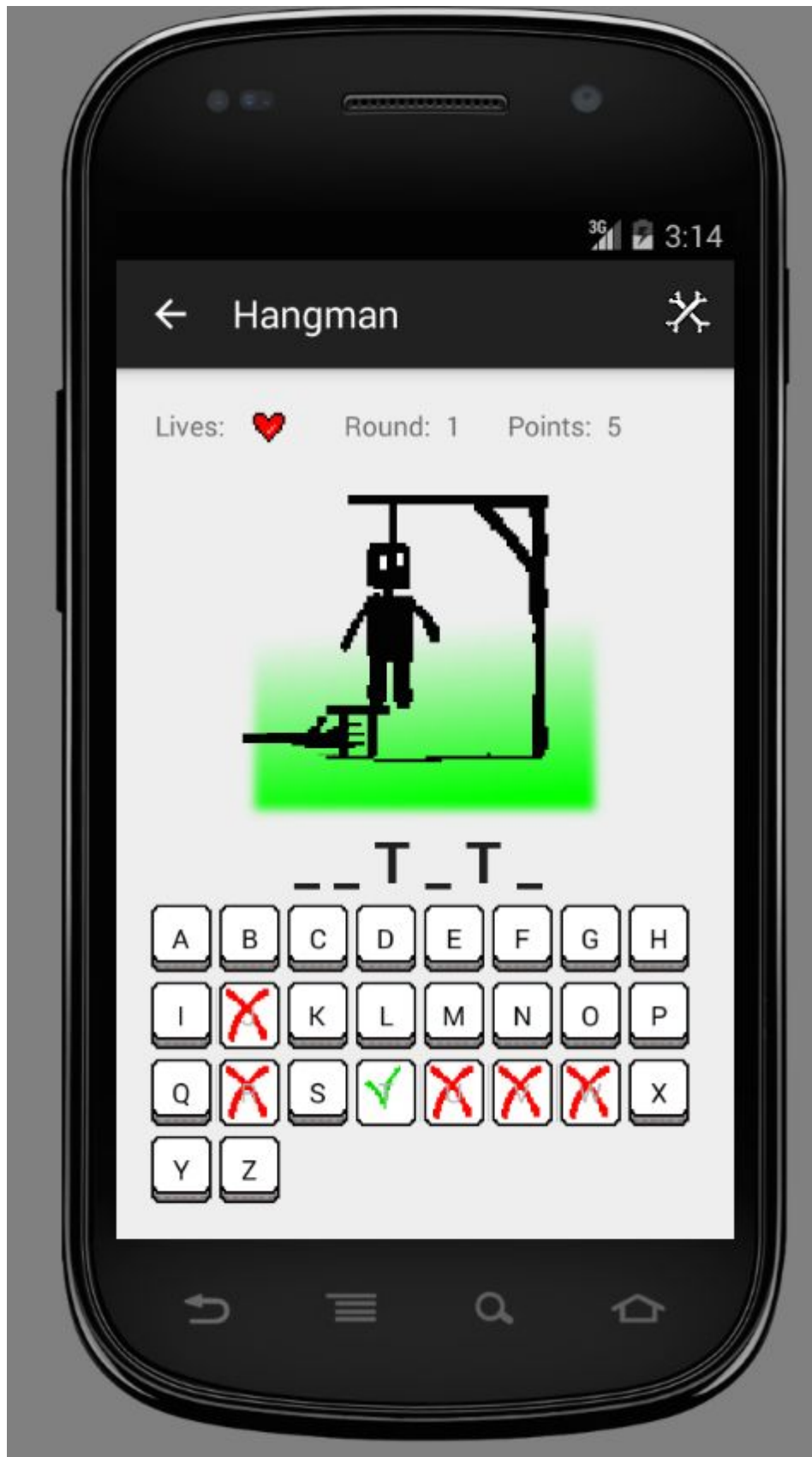
Andre gang du gjetter feil: (Du får tauet rundt nakken)



Tredje gang du gjetter feil: bøddelen strekker ut armen.



Fjerde gang du gjetter feil: Bøddelen tar tak i krakken.

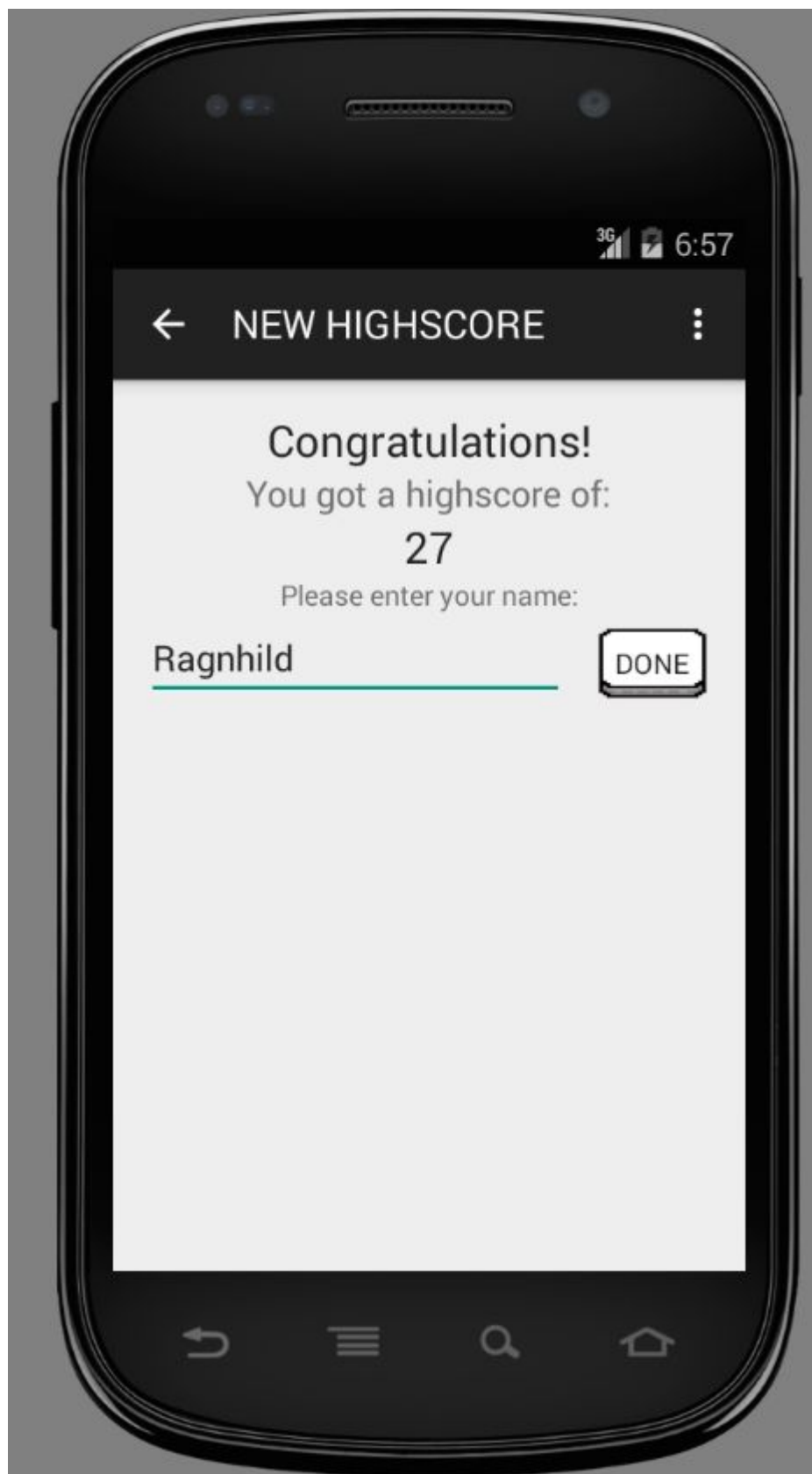


Femte gang du gjetter feil: Bøddelen begynner å dra vekk krakken du står på.



Variant av GameWindow: Sjette gang du gjetter feil: Bøddelen har dratt vekk krakken, og du er død. Trykk "New Game" for å fortsette og "QUIT" for å gå tilbake til hovedmenyen.

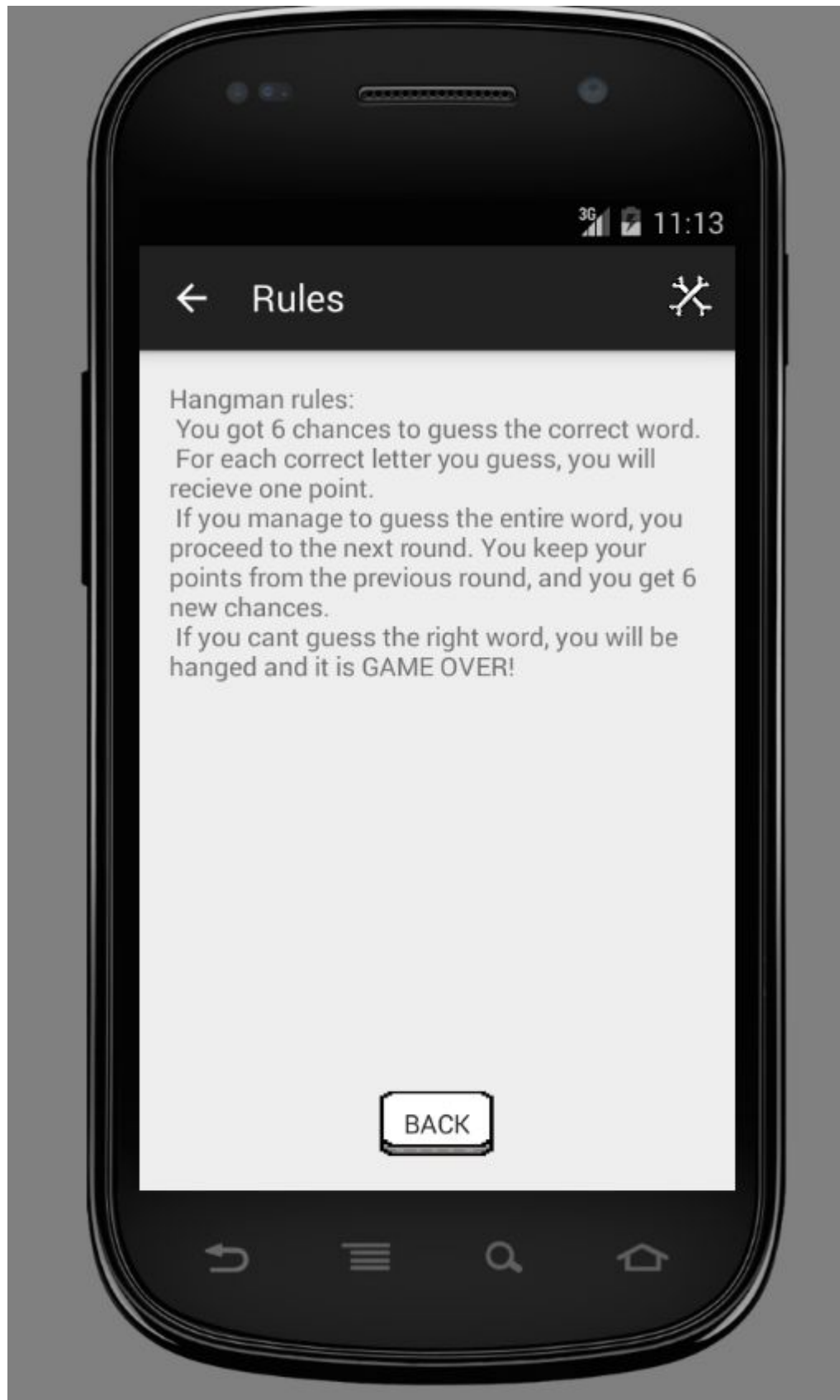




Dersom du ved Game over har slått highscoren eller fått like mange poeng som den, blir du tatt til dette skjermbildet, aktiviteten EnterScore. Her skriver du inn navn, og klikker på "done", så tas du videre til neste skjermbilde:



HighscoreView: viser highscorelista. Ved klikk på back sendes du tilbake til MainActivity.



Skjermбилde 3: Reglene.

## Refleksjon

Hadde jeg hatt bedre tid hadde jeg nok jobbet litt mere med å få highscorelista inn i en tabell istedenfor i et TextView. Og jeg ville fiksa regelsida så den ser litt bedre ut. Samt optimalisert koden min litt og prøvd å finne mer effektive algoritmer. Jeg er generelt fornøyd med denne oppgaven og føler at jeg har lært en del, selv om det har vært stressende og svært tidkrevende, til senere bør jeg lese oppgaveteksten nøyere så jeg ikke gjør for mye.

### KILDER

<https://www.google.com/design/spec/style/icons.html#>

<http://developer.android.com/design/get-started/principles.html>

<http://developer.android.com/training/index.html>

<http://developer.android.com/guide/index.html>