

Do Devs Dream Of Electric Sheep?

A Q/A based Terminal Game that determines
if you're really Human... (sort of)

Inspiration

Inspired by the 1982 film Blade Runner, and Philip K. Dicks novel, “Do Androids Dream of Electric Sheep?”

Based on a machine called the Voight Kampff, from both franchises, that can determine if the subject is a Human or a Replicant - an genetically engineered being identical to Humans. The “interviewer” of the Test, is main character Rick Deckard.

I re-watched both Blade Runner, and Blade Runner 2049 the night before we received the assignment, so I was pretty set on the idea from the start.

Challenges I Faced

Doubting my ability and familiarity with Ruby and OOP definitely hindered creating this app.

Biting off more than I could chew by wanting to implement too many features that wouldn't be achievable in the time frame, given I would've had to do more extensive research.

Utilising Product Management tools to properly plan, structure and create a working app.

How it Works

The app opens with a Welcome Screen, followed by a few introductory questions, before giving the User the option to read a Help and/or Info section, containing how the app works and background information regarding the inspiration behind the app, respectively.

Then the program runs through 12 questions, prompting the User to pick a predefined answer for each.

Finally, after all questions are answered, the program (randomly) places the User into one of three categories;

- Human(Your safe)
- Replicant(Acceptable)
- Rogue Replicant

If you get one of the first two endings, you get a unique message and are then prompted to exit the app.

If you get the “Rogue Replicant” ending, you get a unique message, followed by a screen glitch, signalling that you have been ‘retired’

Who is it for?

The targeted audience is anyone interested in Sci-Fi and/or fans of Blade Runner.

As Artificial Intelligence and the “robots vs humans” theme are common in media, anyone can play the app and understand what’s happening.

This app is purely for enjoyment, based off something I appreciate, and is in no way meant to be offensive if it were to come across that way.

Testing

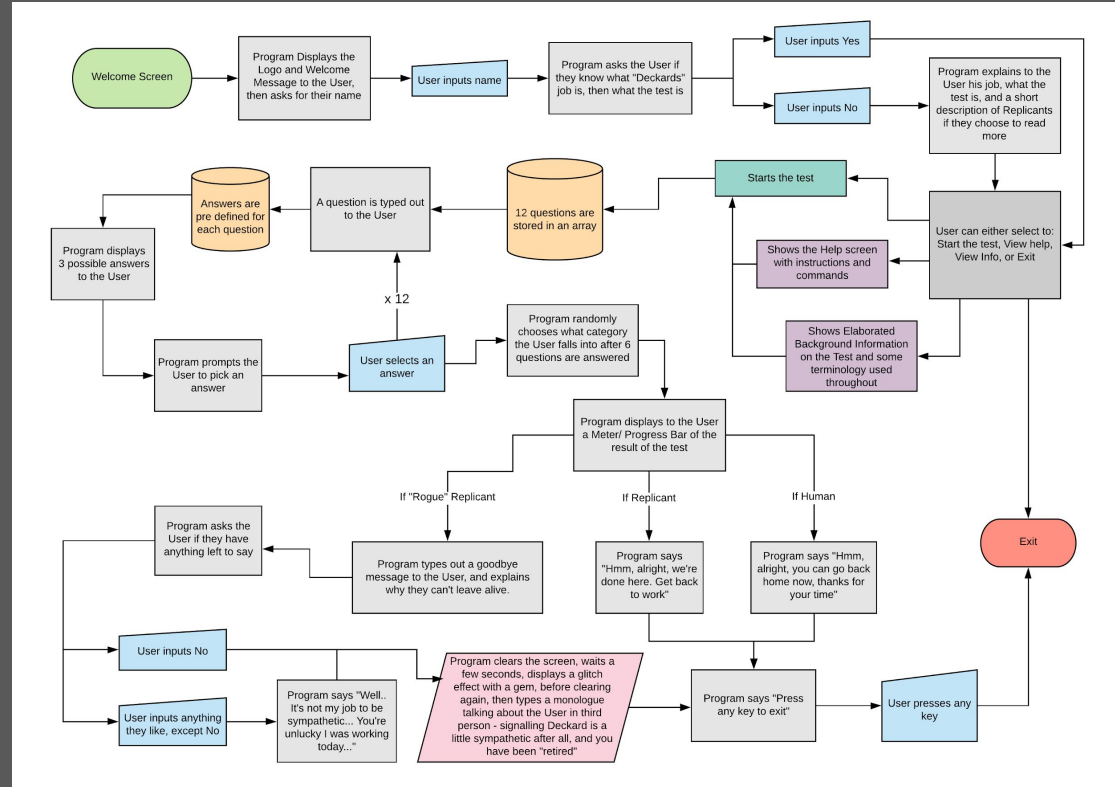
I wasn't the most comfortable with RSpec as I want to be, as I don't think I wrapped my head around it properly, though it's definitely something I'm going to try understand better as I work on my projects.

```
~/CA/assignments/Do_Devs_Dream/src master*  
> rspec spec/start_test_spec.rb  
.question 1  
question 2  
question 3  
.  
  
Finished in 0.00572 seconds (files took 0.12447 seconds to load)  
2 examples, 0 failures  
  
~/CA/assignments/Do_Devs_Dream/src master*  
> █
```

```
~/CA/assignments/Do_Devs_Dream/src/spec master*  
> rspec rand_ending_spec.rb  
F  
  
Failures:  
  
1) RandEnding#user_ending should return a random ending  
Failure/Error: expect(subject.user_ending).to eql(true)  
  
expected: true  
got: nil  
  
(compared using eql?)  
# ./rand_ending_spec.rb:23:in `block (3 levels) in <top (required)>'  
  
Finished in 0.01756 seconds (files took 0.10795 seconds to load)  
1 example, 1 failure  
  
Failed examples:  
  
rspec ./rand_ending_spec.rb:22 # RandEnding#user_ending should return a random ending
```

Flow Control Diagram

This changed a lot throughout the planning and coding process unfortunately, a lot of features that I had to scrap has been put into a future features list that I eventually want to work on in my own time.



Thanks !

Even though, the final outcome wasn't exactly how I envisioned it, I learnt and brushed up on so many things in the last week, and it only makes me excited to create more.

What would I have changed?

Definitely making sure I do research before I start coding, during the planning stage, about what I'm going to need to know in order to accomplish what I wanted to achieve, as I wasted a fair bit of time researching/ learning in the last few days that could have been avoided if my idea was fleshed out much earlier.