GitHub Username: lilggamegenius

Person Generator

Description

This app will enable developers to create fake person info easier to enable them to work on their projects without needing to spend lots of time thinking of fake names.

Intended User

This is for developers who need to generate info for a random person.

Features

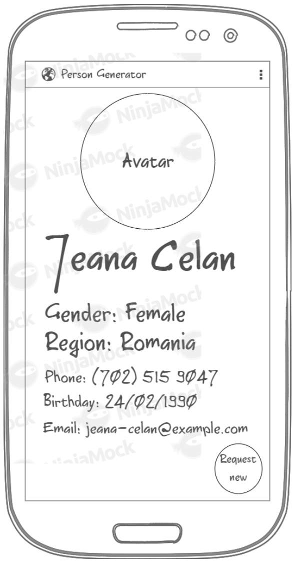
Generate fake info with:

- Age
- Region
- Gender

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1



Jeana Celan

Gender: Female

Region: Romania

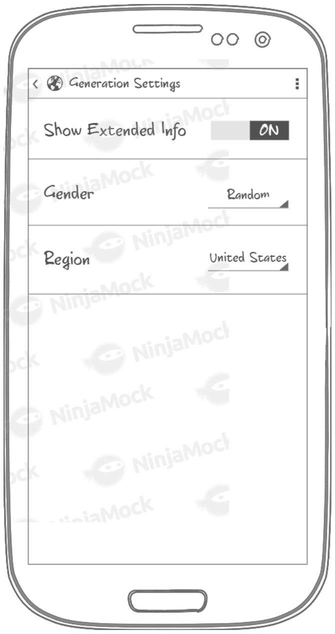
Phone: (702) 515 3047

Birthday: 24/02/1990

Email: jeana-celan@example.com

The main results screen that gives the result

Screen 2



Show Extended Info ON

Gender Random

Region United States

Settings screen to allow filtering results

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

The app will used Shared Preferences as the only data saved are info generation filters

Describe any edge or corner cases in the UX.

For example, how does the user return to a Now Playing screen in a media player if they hit the back button?

Describe any libraries you'll be using and share your reasoning for including them.

Picasso will be used to load the profile image

Describe how you will implement Google Play Services or other external services.

No services will be used

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Create and test classes that will connect to the uinames API

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for MainActivity
- Build UI for Settings
- Build UI for bulk generation

Task 3: Handle Errors

Setup checks to prevent network errors, and handle API errors

Add as many tasks as you need to complete your app.

Submission Instructions

- After you've completed all the sections, download this document as a PDF [File → Download as PDF]
    - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"