Android Developer Test

Test project: CARS

PEAKS SOFTWARE

Introduction

This is a basic programming test that will help us to make a more accurate assessment of your skills and quality level. It tests the ability to read and understand software specifications, to properly implement UI based on a given design and to handle data in an efficient manner.

Quality, quality, quality

At 7 Peaks Software, we take pride in delivering high-quality software.

We have a saying:

"If it has not been tested it doesn't work"

When you deliver this project it should be considered production-ready/ready for deployment.

Estimation

Before development please send project estimation to jobs@7peakssoftware.com
Split estimation into smaller tasks and write them directly in the email.

Example:

- Project setup: 1h
- ...

Deliverables

- Gradle project archive that can be imported into Android Studio
- Assembled apk file

Send deliverables to jobs@7peakssoftware.com

Evaluation criteria

- Estimation is split into suitable parts
- Functioning as described in specifications
- The app looks exactly like in mocks
- The app should not crash or hang, errors should be handled gracefully
- Architecture: App is made in a way that makes it easy to maintain and add new features
- Clean code

Best of luck on the developer test.

Best regards,
Roy Ivar Moe
CTO 7 Peaks Software

Cars - Developer test app

Cars is a simple Android app to test your abilities as an Android developer.

Technical

Supported Android versions: Android 21+
Supported densities: hdpi, xhdpi and xxhdpi
Tablets support: not required
Supported language: English
Package: com.sevenpeakssoftware.**xxx** (use your name instead of **xxx**)

The implementation **must** fully be written in Kotlin and **must** make use of RxJava

UI

You can inspect the UI and download needed resources from Zeplin.

Zeplin project: https://app.zeplin.io/login
Login: android_developer_test
Password: 7p3akssoftware2018

Cars list screen

The Cars list screen is the default screen when opening the app, it displays a list of cars from API.

Each car item in the list consists of the following elements:

- Image
- Title
- Publish date
- Ingress

Dates should use next format:

- If within current year: “[day] [month], [time]”. For example: “16 March, 12:34”
- If within different year: “[day] [month] [year], [time]”. For example: “20 December 2017, 21:43”

Time should follow system settings for 12h / 24h format, for example, “09:43 PM” vs “21:43”.

Offline support

The content should be available offline. That means it should be possible to view Cars list screen without an internet connection. Note, that items should still be updated from API if internet connection is available, use cached data as a fallback in case of any problems getting fresh data from the backend.

Use a database for offline storage.

API

The app uses a simple API.

Base URL: https://www.apphusetreach.no/application/119267

Get all articles

Request	GET [base_url]/article/get_articles_list			
Response	HTTP 200			
	Body	status	String	Response status
		content	Array of Articles	List of news articles

Article

Field	**Type**	**Comment**
id	Int64	Unique id
title	String	Title
ingress	String	Ingress text
image	String	Image URL
dateTime	String	Date in format “dd.MM.yyyy HH:mm” using UTC time zone.
tags	Array of String	Associated tags
content	Array of Item	List of content items for the article
created	Int64	Creation timestamp (Unix time)
changed	Int64	Changed timestamp (Unix time)

Item

Field	**Type**	**Comment**
type	String	Item type. Currently only supported type is “text”
subject	String	Title
description	String	Text