

## CMX Training Challenges

**Contexto:** muitas vezes os novos estagiários e novos integrantes que chegam ao CMX possuem um baixo nível de conhecimento específico sobre Android. Tanto a USP quanto UFSCar e IFSP possuem cursos que desenvolvem o raciocínio lógico do aluno e oferecem a base para estes adquirirem novos conhecimentos quando entrarem no mercado de trabalho. Porém, não desenvolvem habilidades específicas, sejam elas habilidades voltadas para desenvolvimento web, Android ou iOS, arquitetura de software, design patterns, banco de dados noSQL ou qualquer outra tecnologia mais recente que esteja sendo usada na área de desenvolvimento.

**Proposta inicial:** levando em consideração o contexto atual dos novos integrantes e também o desenvolvimento da equipe, a ideia é elaborar um cronograma de desafios para os novos integrantes da equipe CMX e integrantes que queiram desenvolver seus conhecimentos sobre componentes, bibliotecas e padronizações Android.

**Como funciona:** os estagiários trabalham aproximadamente 30 horas por semana e normalmente estudam no período da noite, portanto, não possuem muitas horas vagas para desenvolver projetos pessoais e aprender novas tecnologias em suas casas. A ideia do cronograma de desafios é utilizar parte do horário de trabalho que estejam mais “tranquilos” para desenvolver os desafios.

**Os desafios:** cada desafio será desenvolvido em um determinado tempo (a ideia inicial é de duas semanas para cada desafio) e, para cada desafio, o integrante irá escolher uma quantidade mínima de tópicos que deverão estar contidos no projeto. O desafio basicamente é desenvolver um aplicativo à escolha de cada participante, ou seja, cada participante irá elaborar a ideia do seu aplicativo utilizando uma única regra: deverá conter um número mínimo de tópicos. Os tópicos serão divididos em três níveis baseados no nível de dificuldade de utilizar cada tópico.

**Por exemplo:** *Desafio 1: o projeto deste desafio deverá utilizar no mínimo três tópicos nível 1.*

Nível 1	Nível 2	Nível 3	Nível Extra
Retrofit	Recycler View	Custom View	Kotlin
Shared Preferences	List View	Dagger 2	Android Annotations
Fragments	Navigation Drawer	RxJava	Android Build Flavours
SQLite	Tabs	Animations	Firebase
Multi Language App	Google Maps	Services	MVP design pattern

*\* Esta tabela foi desenvolvida levando em consideração a dificuldade e prioridade para aprender sobre o assunto*

### Links interessantes:

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

<https://br.udacity.com/course/material-design-for-android-developers--ud862/>

<https://br.udacity.com/course/google-location-services-on-android--ud876-1/>

<https://br.udacity.com/course/how-to-create-anything-in-android--ud802/>

<https://br.udacity.com/course/firebase-essentials-for-android--ud009/>