

Android/Kotlin – ViewModel

O **ViewModel** é a principal entidade da camada de lógica de negócio da arquitetura recomendada pelo Google Android.

É nessa camada que há as invocações à APIs de busca de dados, camada de modelo, há também inicializações de algumas classes de domínio, principalmente as que têm atualizações que devem ser refletidas na camada de visualização.

O máximo possível a se obter com a camada de lógica de negócio não vem somente com o uso da API **ViewModel** e sim com um conjunto dela com a API **LiveData**, está última que seria equivalente a APIs de react como: RxJava ou Agera.

A principal característica do **ViewModel** é a capacidade de manter dados em memória enquanto, por exemplo, há uma reconstrução da [atividade](#) a qual ele está vinculado.

Exemplo de classe **ViewModel**:

```
class SignUpViewModel: ViewModel() {
```

```
    val usuario: User
```

```
    init {  
        usuario = User()  
    }  
}
```

O acesso a instância da classe acima, direto de uma atividade:

```
class SignUpActivity : AppCompatActivity() {
```

```
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_sign_up)
```

```
        val model = ViewModelProviders.of( this ).get( SignUpViewModel::class.java )  
    }  
}
```

Apesar de não estar explícito na documentação, nas classes **ViewModel** têm de haver um construtor vazio, sem parâmetros.

Por hoje é só pessoal, sucesso nos estudos.