FOSSIL
GROUP

Fossil Techtalk

Fossil Vietnam × FIT - Người trong nghề

Introduction to Mobile Architecture

21 April, 2022

Fossil Techtalk

FOSSIL
GROUP

SPEAKERS

Fossil Techtalk

Thai Huynh

Senior Cloud Engineer

- **Bachelor in Computer Engineering, VNU-HCMUT**
- **Former Mobile Solution Architect at Fossil (2018-2019)**
- **Responsible for:**
  - Finding the best solution to solve the business problems.
  - Describing the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- **Experience of working with different many frontend platforms:**
  - Desktop (Window/Linux)
  - Mobile (Android, iOS, Flutter)

SPEAKERS

Fossil Techtalk

Ngo Huynh

Senior App Development Engineer

- **8 Years working experiences of mobile application programming**
- **Responsible for:**
  - Maintain and developing the Android/iOS app
  - Monitoring updates and possible security threats
  - Troubleshoot problems and ensuring the best performance
  - Unit, UI, Integration tests and setup CI/CD
  - R&D new technology and build Proof of concept (POC)
- **Experience of working with different native and cross platforms:**
  - iOS, Android
  - React Native
  - Flutter

TABLE OF CONTENTS

WHAT IS ARCHITECTURE?

Wiki

- Making fundamental structural choices
- Discipline of creating such structures

Popular

- Mobile: MVC, MVP, MVVM, VIPER,...
- Web: Flux, Redux,...
- Flutter: BLoC,...

WHY IS ARCHITECTURE IMPORTANT?

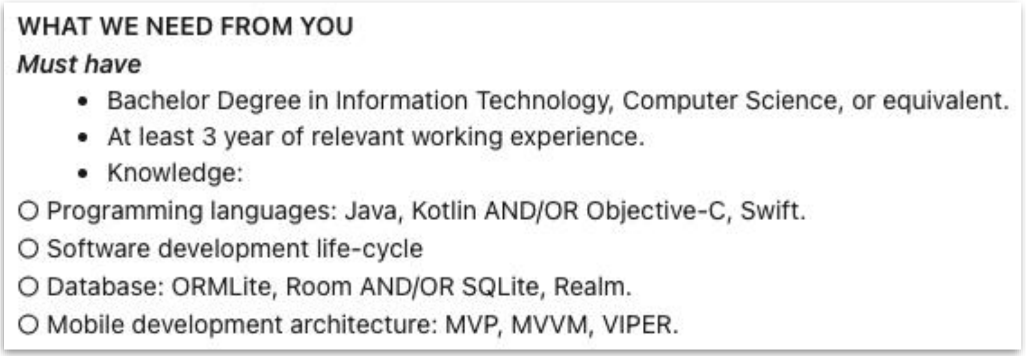
Go far of go fast?

- Stability
- Extendable

Solo or Teamwork?

- Align
- Parallels

What employers need from you?



WHAT WE NEED FROM YOU

Must have

- Bachelor Degree in Information Technology, Computer Science, or equivalent.
- At least 3 year of relevant working experience.
- Knowledge:

○ Programming languages: Java, Kotlin AND/OR Objective-C, Swift.
○ Software development life-cycle
○ Database: ORMLite, Room AND/OR SQLite, Realm.
○ Mobile development architecture: MVP, MVVM, VIPER.

HOW DOES ARCHITECTURE HELP FOSSIL?

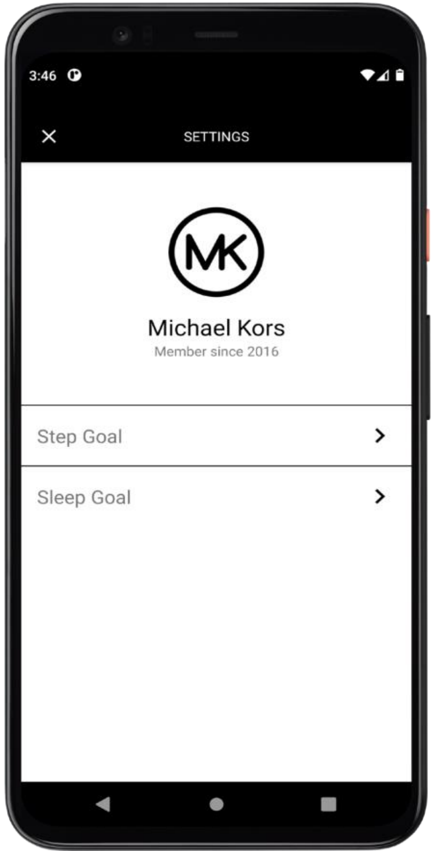
When

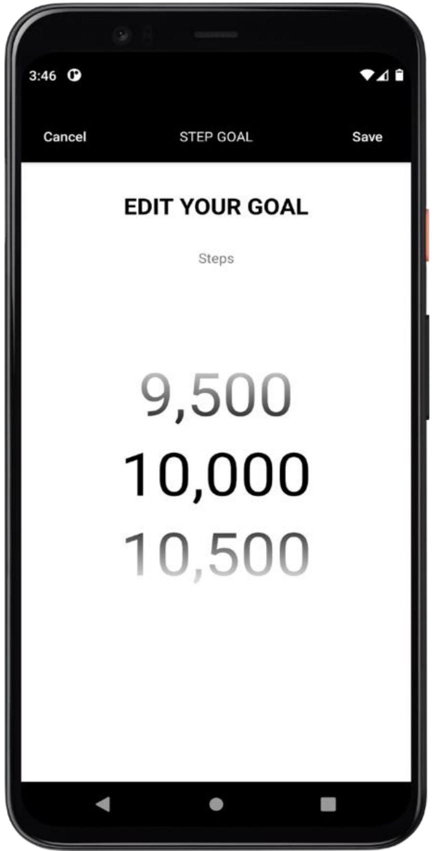
- Using MVC architecture first
- Fossil delivers 26 apps for 13 brands.

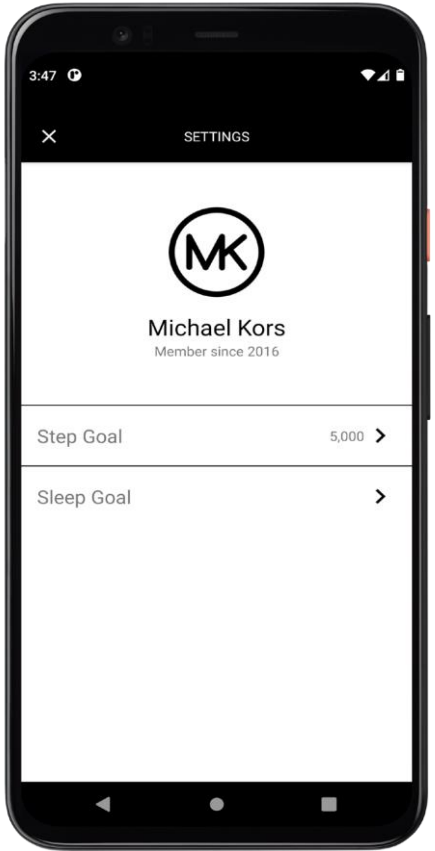
Then

- Reuse code as much as possible.
- Separate View component.
- Easy to write tests.

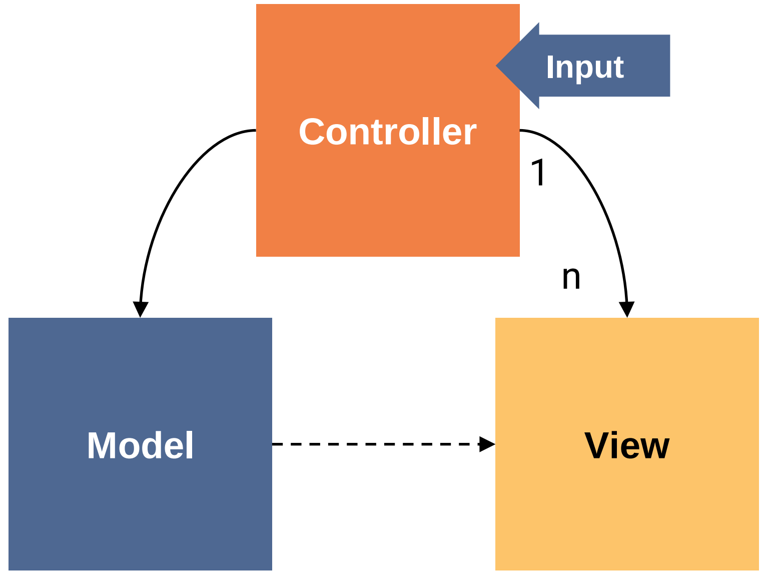
HOW DOES ARCHITECTURE HELP FOSSIL?







MVC



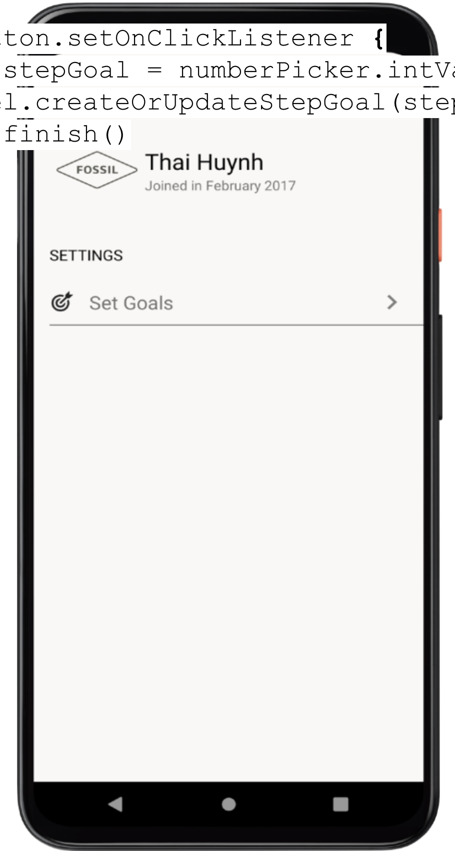
```
@AndroidEntryPoint
class StepGoalActivity : AppCompatActivity() {
    @Inject
    override fun onCreate(savedInstanceState: Bundle?) {g
        binding.lifecycleOwner = thisckListener {
    }verride fun onCreate(savedInstanceState: Bundle?) {
        binding.saveButton.setOnClickListener {w(this,
    override fun onResume() {nding.stepGoalPicker.intValue()
        super.onResume()OrUpdateStepGoal(stepGoal) {tValue()
        binding.model = model
    }        }ng.stepGoalPicker
}          }
   }
}
<layout >
    <data>pGoalModel {
        <variable
            name="model"  Int? = null
type="com.fossil.architecture.example.model.StepGoalModel"
/>  val stepGoalLiveData = MutableLiveData<Int?>()
    </data>
    fun createOrUpdateStepGoal(newStepGoal: Int, listener:
    <com.fossil.architecture.example.widget.NumberPicker
            android:id="@+id/stepGoalPicker"
            android:value="@{model.stepGoalLiveData}" />
</layout>stener?.invoke(newStepGoal)
    }
}
```

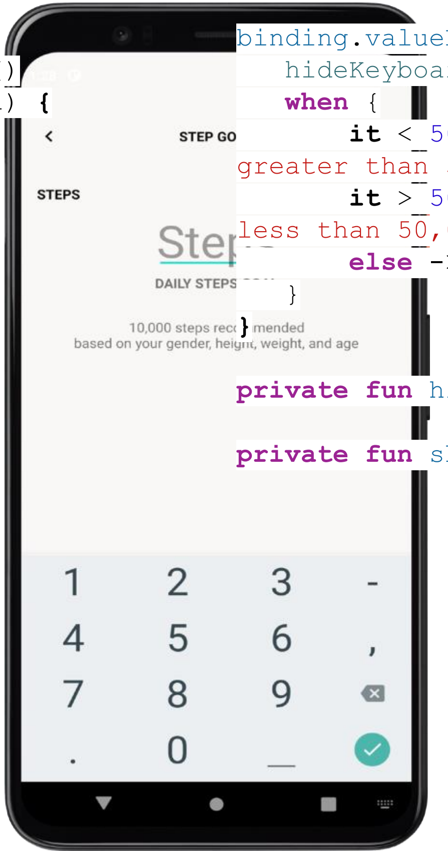
+ Simple to approach
- Testability
- Maintenance

HOW DOES ARCHITECTURE HELP FOSSIL?

Fossil Techtalk

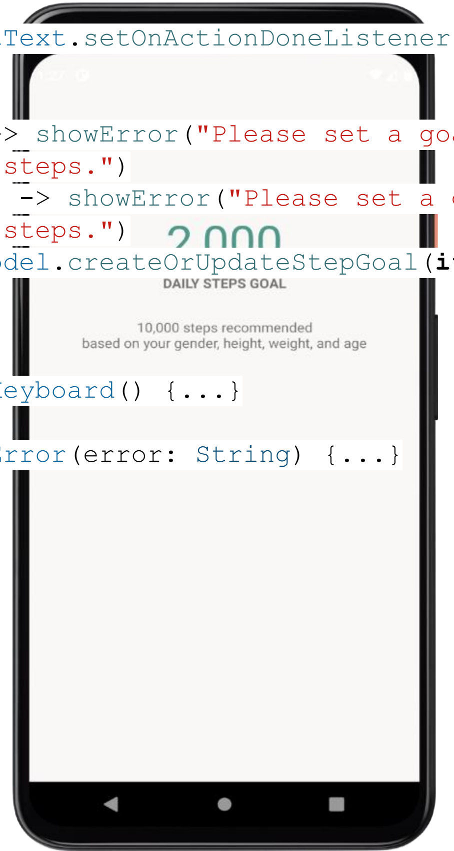
```
saveButton.setOnClickListener {
    val stepGoal = numberPicker.intValue()
    model.createOrUpdateStepGoal(stepGoal) {
        finish()
    }
}
```





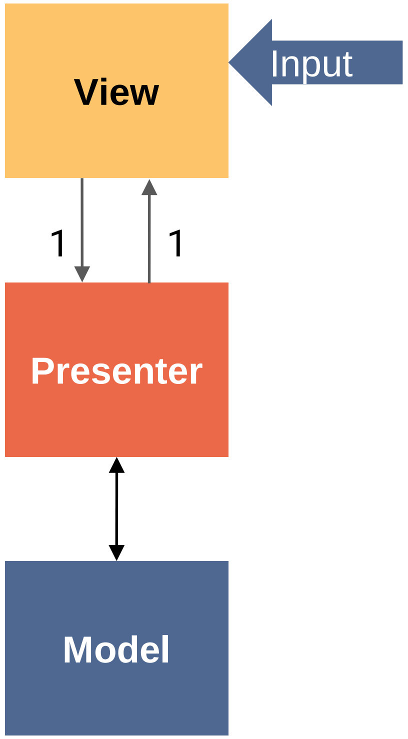
10,000 steps rec mended
based on your gender, heig , weight, and age

```
binding.valueEditText.setOnActionDoneListener {
    hideKeyboard()
    when {
        it < 500 -> showError("Please set a goal of
greater than 500 steps.")
        it > 50000 -> showError("Please set a goal of
less than 50,000 steps.")
        else -> model.createOrUpdateStepGoal(it)
    }
}

private fun hideKeyboard() {...}

private fun showError(error: String) {...}
```



10,000 steps recommended
based on your gender, height, weight, and age

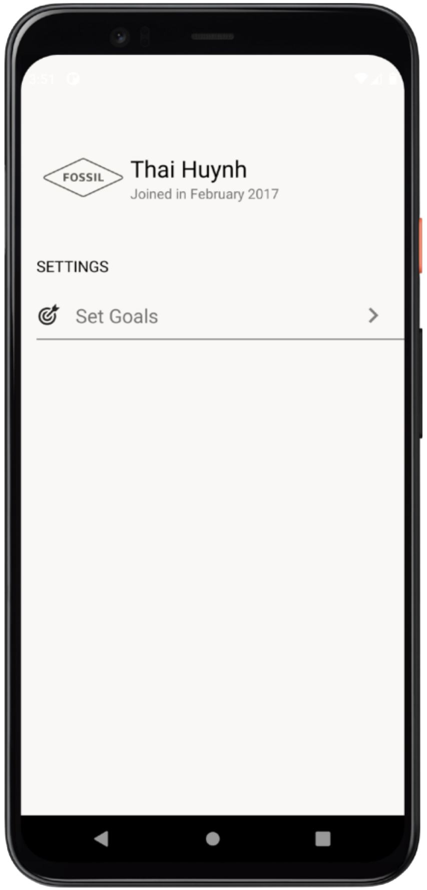
MVP



```
@AndroidEntryPoint<T> {: Fragment(), SetGoalsContract.View {
class SetGoalsActivity : AppCompatActivity() {
   @Injecte fun onViewCreated(view, __) {
   lateinit var model: StepGoalModelr {
interface BasePresenter {}.finish()
   override fun onCreate(savedInstanceState: Bundle?) {
       var view = SetGoalsFragment.newInstance()
       val presenter = SetGoalsPresenter(view, model)
       view.setPresenter(presenter)
  }nterface View : BaseView<Presenter> {
}   fun showStepGoal(stepGoal: Int?)..")
   }      else -> presenter.createOrUpdateStepGoal(it)
class SetGoalsPresenter(
   private val view: SetGoalsContract.View,
   private val model: StepGoalModelewStepGoal: Int)
) : SetGoalsContract.Presenter {
   override fun createOrUpdateStepGoal(newStepGoal: Int) {
       model.createOrUpdateStepGoal(
           newStepGoal,r(error: String) {...}
           view::showStepGoal
       )
   }

   override fun start() {
       model.getStepGoal(view::showStepGoal)
   }
}
```

+ Clean
+ Testability
+ Maintenance
- Boilerplate code
- Presenter - View 1-1 relationship

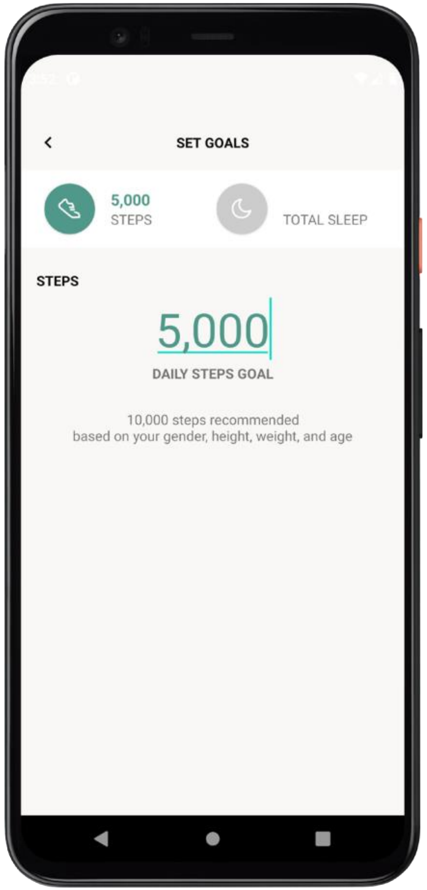
HOW DOES ARCHITECTURE HELP FOSSIL?



Thai Huynh
Joined in February 2017

SETTINGS

Set Goals



SET GOALS

5,000
STEPS

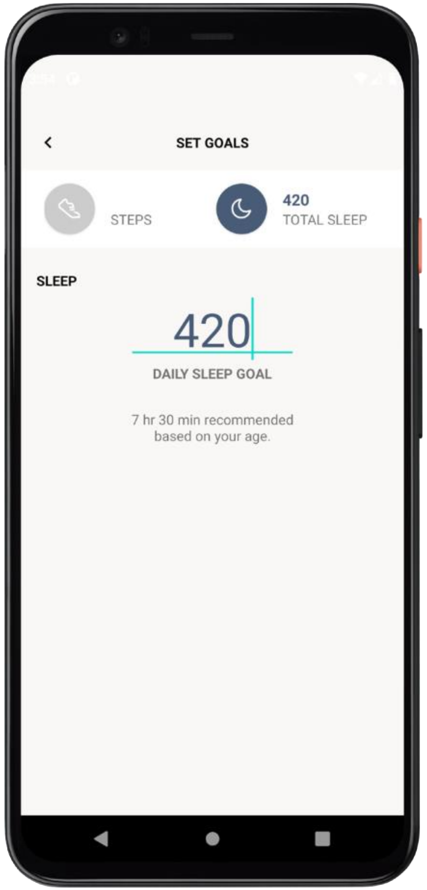
TOTAL SLEEP

STEPS

5,000

DAILY STEPS GOAL

10,000 steps recommended
based on your gender, height, weight, and age



SET GOALS

STEPS

420
TOTAL SLEEP

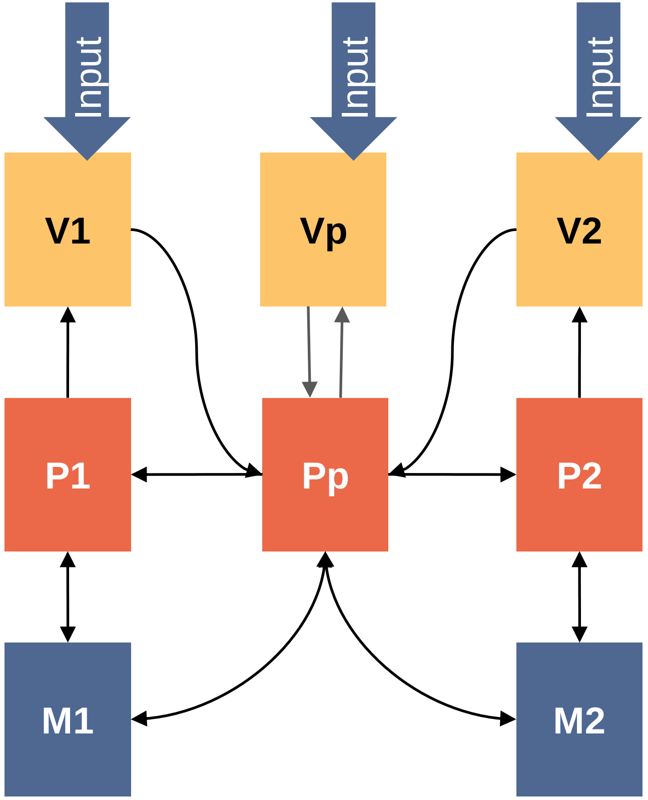
SLEEP

420

DAILY SLEEP GOAL

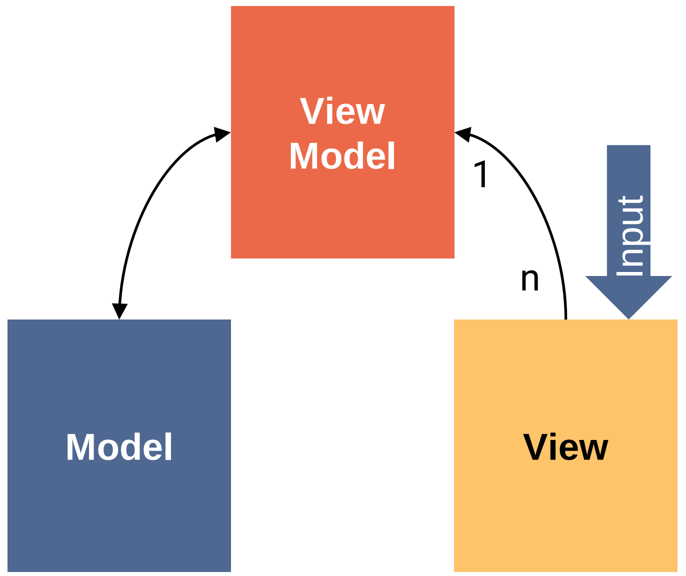
7 hr 30 min recommended
based on your age.

MVP



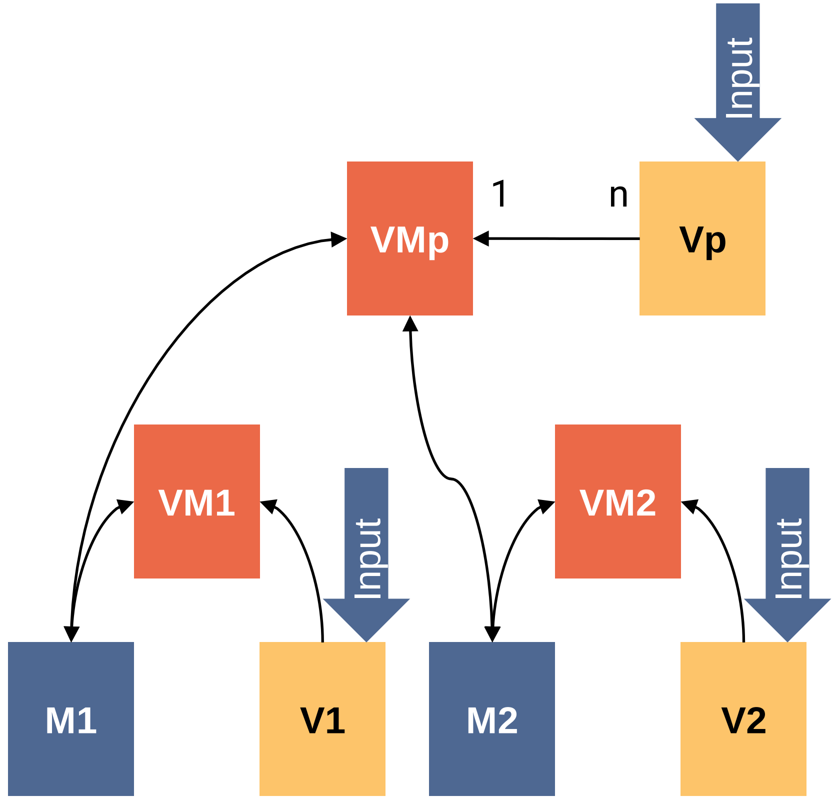
```
@AndroidEntryPointenter(ct {
class SetGoalsFragment : Fragment(), SetGoalsContract.View {
   @Inject val stepGoalModel: StepGoalModel,
   lateinit var stepGoalModel: StepGoalModel1
) : SetGoalsContract.Presenter {
   @Inject
   lateinit var sleepGoalModel: SleepGoalModel
       stepGoalPresenter: StepGoalContract.Presenter,
   private var stepGoalView: StepGoalFragment? = nullr
   private var sleepGoalView: SleepGoalFragment? = null
           stepGoalPresenter: StepGoalContract.Presenter,
   override fun onViewCreated(view: View, ntract.Presenter
savedInstanceState: Bundle?) {al {
      val stepGoalPresenter = t)
StepGoalPresenter(stepGoalView!!, stepGoalModel)
      val sleepGoalPresenter =
SleepGoalPresenter(sleepGoalView!!, sleepGoalModel)
      stepGoalView!!.setPresenter(presenter)
      sleepGoalView!!.setPresenter(presenter)
      presenter.setPresenters(stepGoalPresenter,
sleepGoalPresenter)
   }
}
```

MVVM



```
@HiltViewModeloint
class SleepGoalViewModel @Inject constructor(private val
model: SleepGoalModel) : ViewModel() {
   val sleepGoal = model.sleepGoalLiveData.map { odel
it?.toString() ?: "" }
   private var binding by
   fun createOrUpdateSleepGoal(newSleepGoal: Int) {
       model.createOrUpdateSleepGoal(newSleepGoal)
   }verride fun onViewCreated(view: View,
}avedInstanceState: Bundle?) {
       viewModel =
<layout >Provider(this)[SleepGoalViewModel::class.java]
       binding.viewModel = viewModel
   <data>nding.lifecycleOwner = this
       binding.sleepGoalEditText.setOnActionDoneListener {
       <variableeyboard()
           name="viewModel"OrUpdateSleepGoal(it)

type="com.fossil.architecture.example.ui.sleepgoal.SleepGoal
ViewModel" />
   </data> fun hideKeyboard() {...}
}
   <com.fossil.architecture.example.widget.MyEditText
       android:id="@+id/sleepGoalEditText"
       android:text="@{viewModel.sleepGoal}"/>
</layout>
```

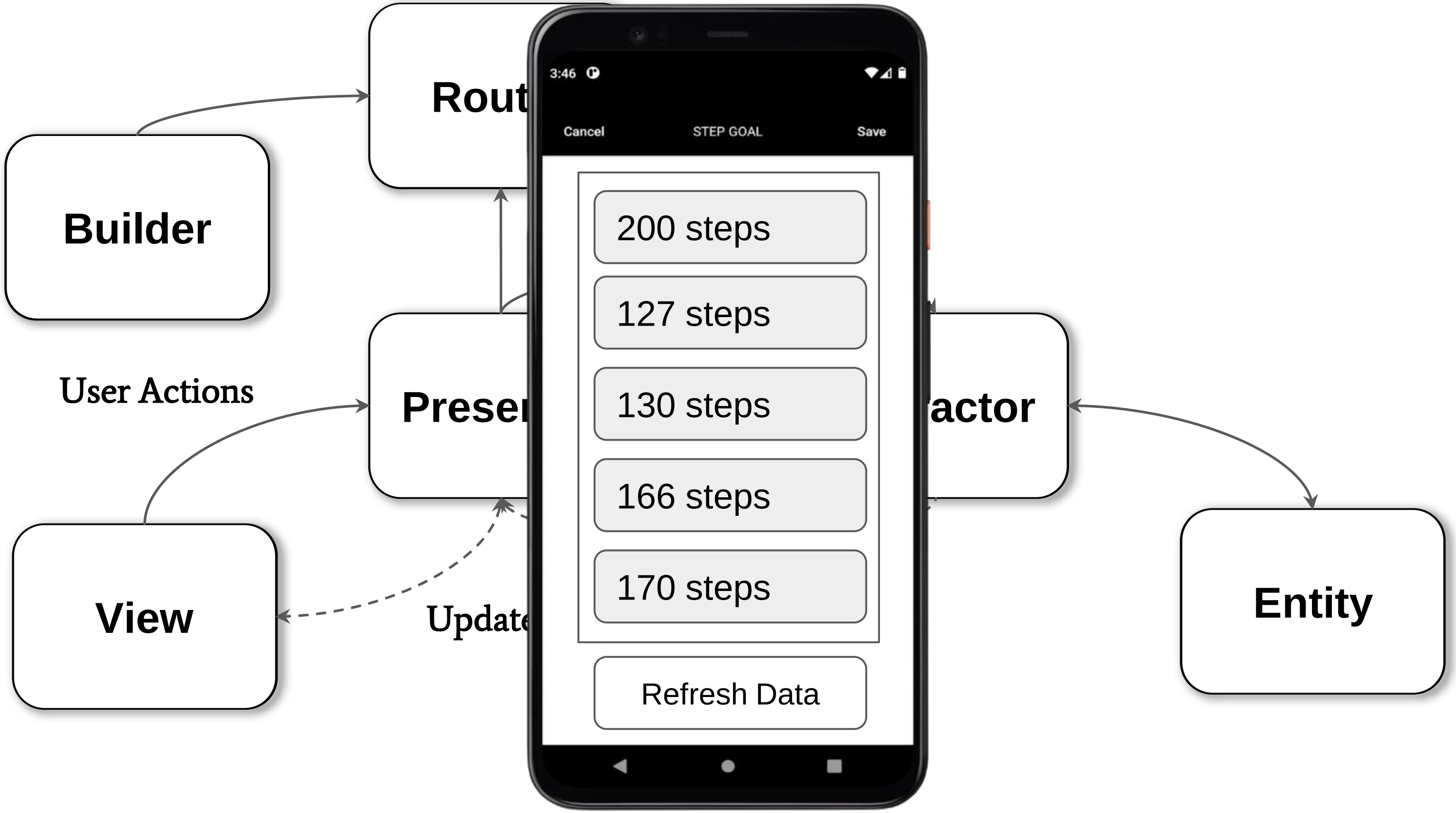
MVVM



```
@HiltViewModel
class SetGoalsViewModel @Inject constructor(
    stepGoalModel: StepGoalModel,
    sleepGoalModel: SleepGoalModel
) : ViewModel() {
    val stepGoal = stepGoalModel.stepGoalText
    val sleepGoal = sleepGoalModel.sleepGoalLiveData.map {
it?.toString() ?: "" }
}

<layout>

  <data>

      <variable
          name="viewModel"

type="com.fossil.architecture.example.ui.setgoals.SetGoalsVi
ewModel" />
    </data>

    <TextView
        android:id="@+id/stepGoal"
        android:text="@{viewModel.stepGoal}" />

    <TextView
        android:id="@+id/sleepGoal"
        android:text="@{viewModel.sleepGoal}" />
</layout>
```

+ Less code
+ Testability
+ Business code doesn't know View
- Difficult to approach

VIPER



3:46

Cancel STEP GOAL Save

200 steps

127 steps

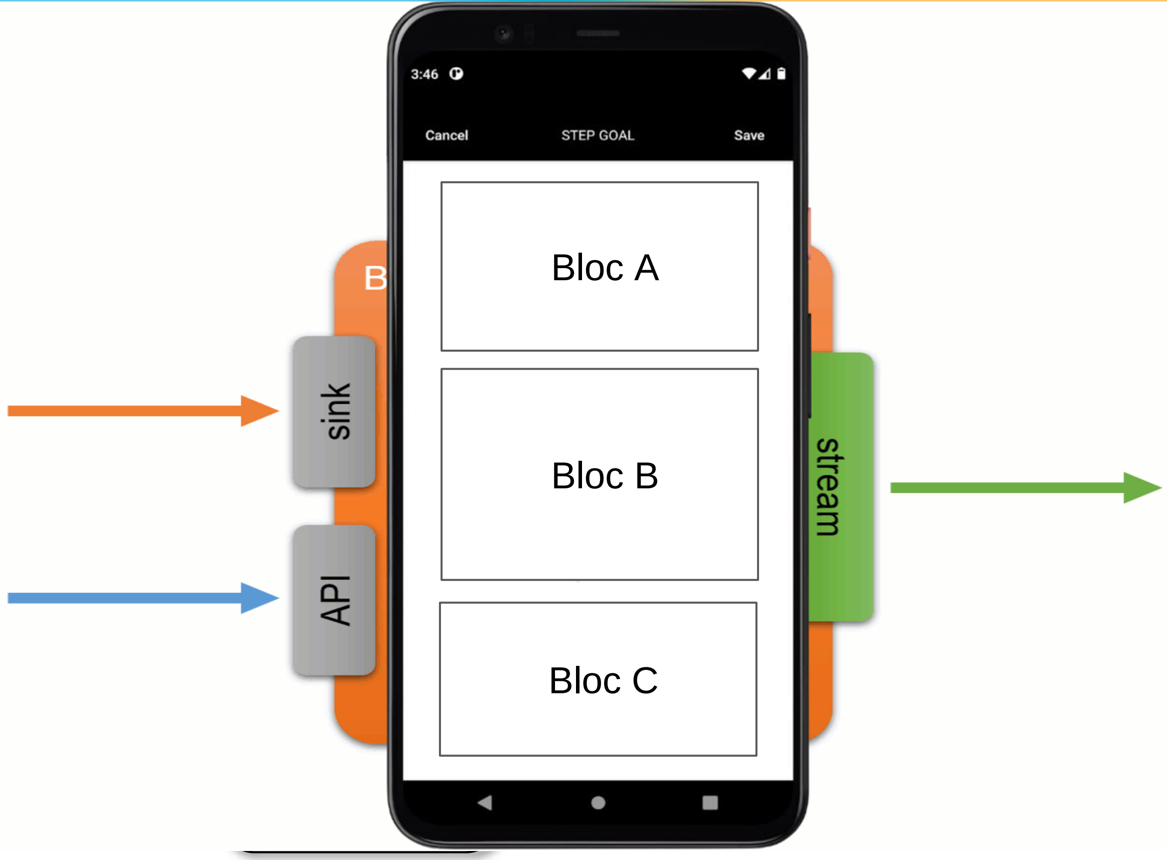
130 steps

166 steps

170 steps

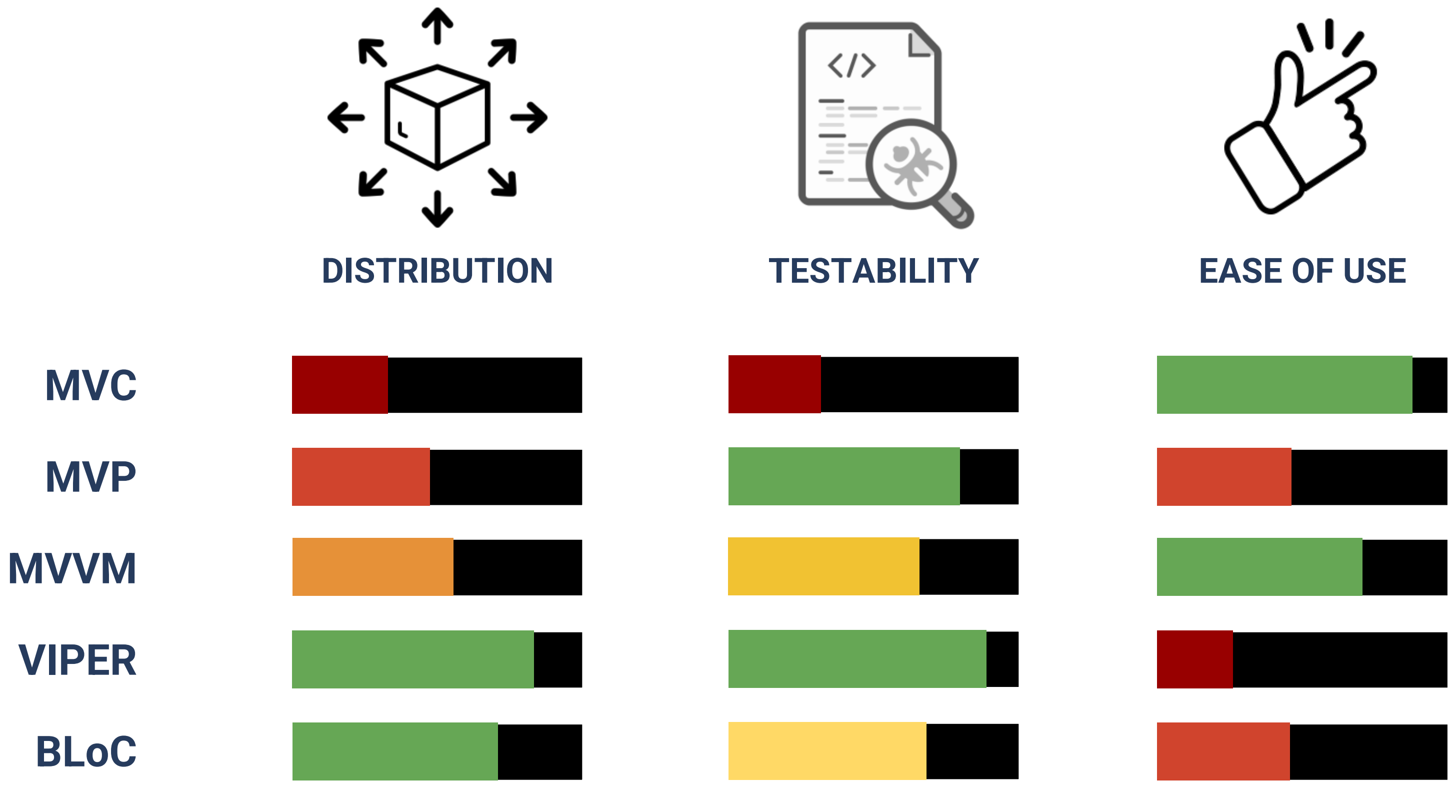
Refresh Data

BLoC

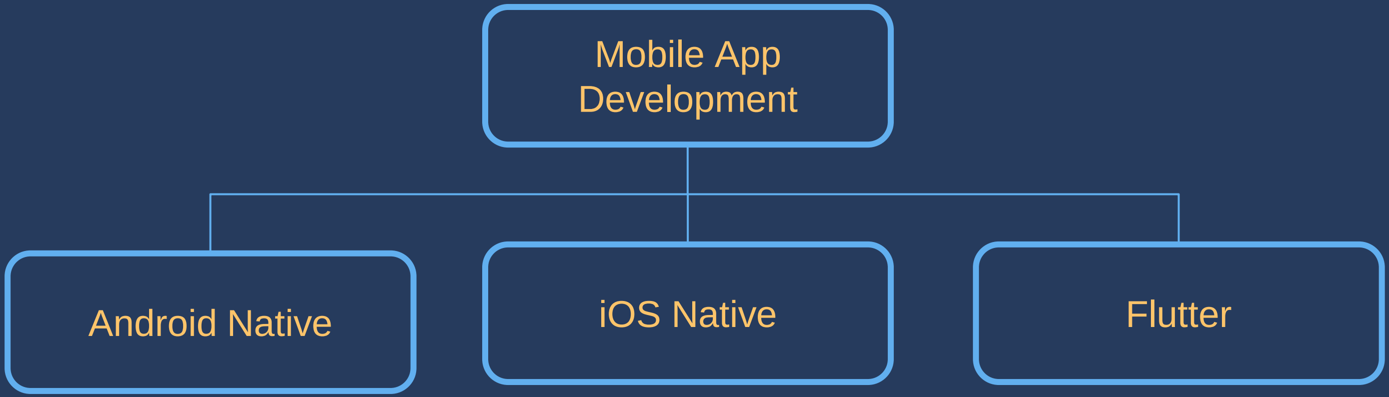

3:46
Cancel
STEP GOAL
Save
Bloc A
Bloc B
Bloc C
B
sink
API
stream

QUICK COMPARISON

Fossil Techtalk



MOBILE APP DEVELOPMENT TEAM SCOPE



WHAT WE NEED FROM YOU?

Fossil Techtalk

- Experience with one of the modern native mobile development languages: Swift, Kotlin, Java
- Experience working with Android or iOS SDK
- Experience with cross-platform development framework such as Flutter is a plus
- Solid understanding of version control principles
- Good understanding of OOP
- Knowledge of data structures and algorithms (nice to have)

WE'RE LOOKING FOR MOBILE DEVELOPMENT INTERNS!

VIEW JOB DESCRIPTION

Facebook: Fossil Vietnam Careers

LinkedIn: Fossil Vietnam

Recruitment Email: people@fossil.com

FOSSIL GROUP

Thank you for Joining Us

Fossil Techtalk