

Android basics with Compose



Index

basetis

- Who are we?
- Success stories



Android

- Introduction
- Development

Methodology and Tools

- Scrum
- Tools

Live Coding!

basetis

basetis: who are we?

We are a team of **people** with a strong will for **social transformation** that perform services in the field of **information technologies** and **consulting**.

In Basetis, we think of people, we develop trust, we commit ourselves to realizing the best services and contribute to social improvement.

Created in 2009, Basetis's main areas of action are data management, software and web programming and development, mobility solutions, business, systems administration and design.



basetis: who are we?

We base our organizational model on the **TEAL** paradigm.

SELF-MANAGEMENT

- Distributed intelligence.
- Anyone can make any decision.
- Asking for advice is mandatory.
- Responsibility for meritocracy.
- Communication and transparency.



COMPLETENESS

- Each person must be able to show themselves as they are.
- The environment must be flexible and facilitate the integration of diversity.
- Authenticity is basic in every way.
- Equal opportunities for all genders, backgrounds or sexual identities.
- Happily productive and productive.



EVOLUTIONARY PURPOSE

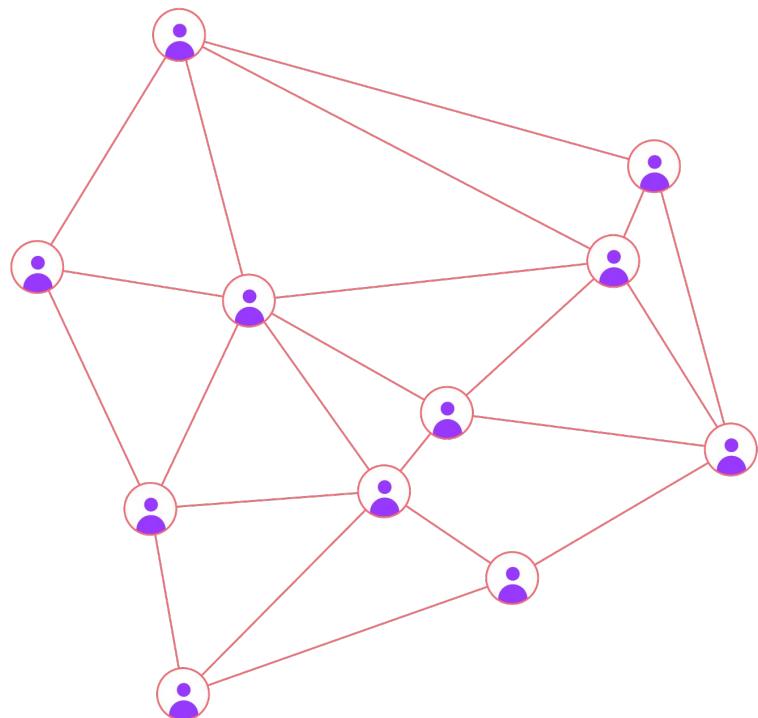
- A company is not just a business.
- We identify with the vision of the Economy of the Common Good.
- Commitment to society, in all dimensions.
- Over time, everything evolves; also the company



basetis: who are we?

Mobile Team Overview

- 350 People at Basetis → 30 at Mobile Team
- 8 with + 5 years XP at the Mobile Sector
- **Expertise:** Android, iOS, Hybrid
- Technologies: **Kotlin**, Java, **Swift**,
Swift UI, Obj-C, **Flutter**, Kotlin Multiplatform
- + 80 Apps of Experience at Basetis
- *Developing Amazing Apps since 2013*



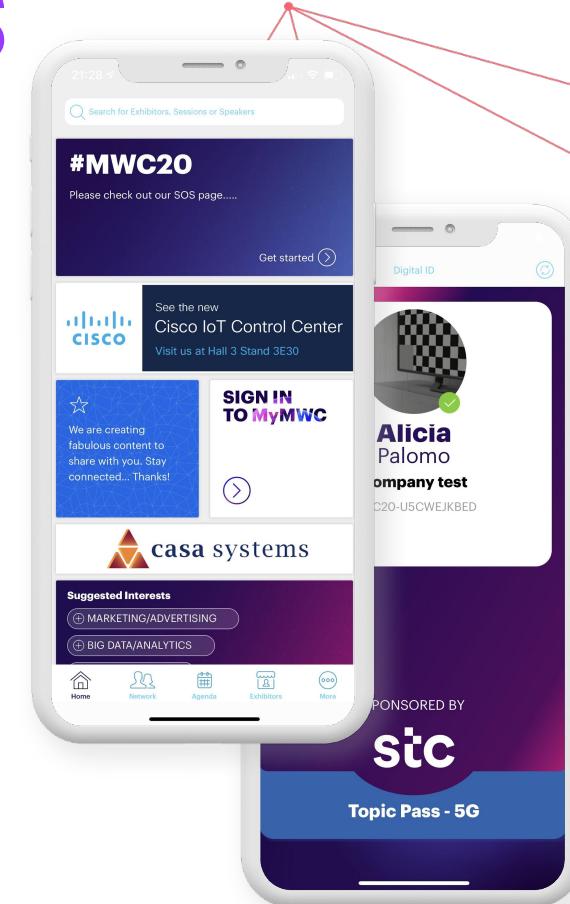
basetis: success stories

Success Stories

Mobile World Congress

MWC, QUALITY AND IMPROVEMENT

Basetis has managed the Mobile World Congress application since the 2020 edition, improving and optimizing processes and creating new functionalities.



basetis: success stories

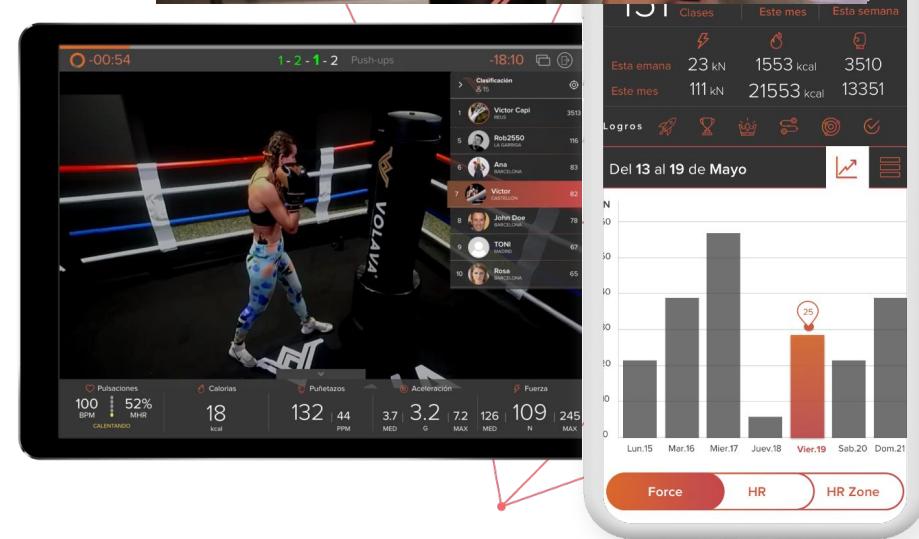
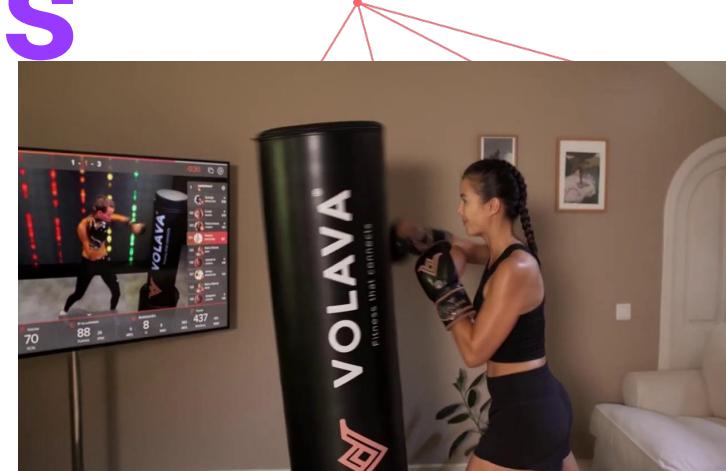
Success Stories

Volava

FITNESS CONNECTION

Fitness class application connected to physical devices via bluetooth (spinning bike, treadmill, sensors in boxing gloves).

Detection of blows and different movements to an AI system.



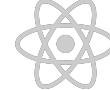
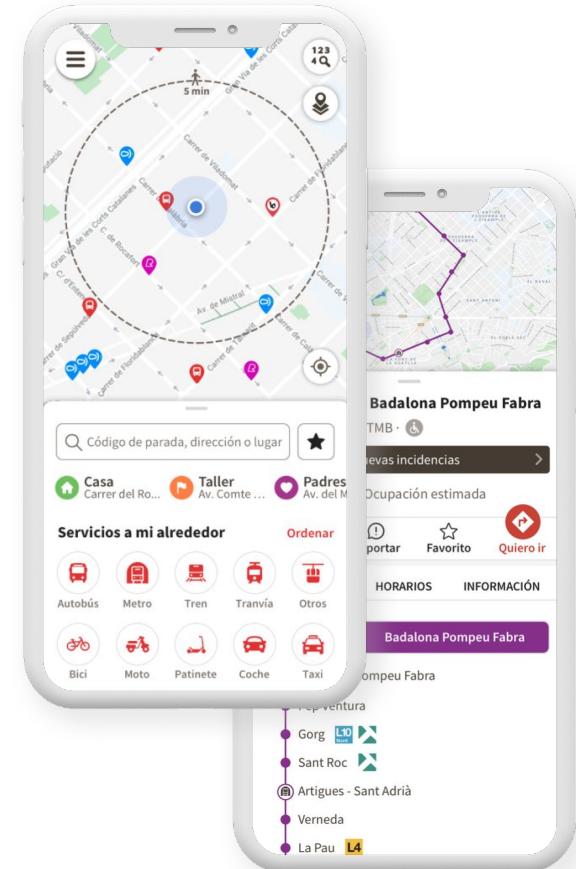
basetis: success stories

Success Stories

AMB Mobilitat

THE FUTURE OF MOBILITY

Project currently under development, is the ambitious renovation of AMB's flagship application, which encompasses all public transport in the Barcelona area.



Android

What is Android?

- It's an Operative System with a Linux shell, contains Open Source code.
- Developed by Android Inc. until Google's acquisition Google in 2005
- Introduced by Google and the Open Handset Alliance in 2007
- Some of the source code is distributed as open-source within the Android Open Source Project (AOSP)



android

Programming languages

- Java (version 11 since Android 13, 8 on previous versions)
- Kotlin



In 2017, Google announced **official support** to Kotlin.

Since 2019 Google recommends Kotlin as the **main language** to be used for Android development.

Android: development

Java - classes

```
public class Person {  
    private String name;  
    private int age = 0;  
  
    public Person(String name, int  
age) {  
        this.name = name;  
        this.age = age;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String  
name) {  
        this.name = name;  
    }  
  
    public int getAge() {  
        return age;  
    }  
  
    public void setAge(int age) {  
        this.age = age;  
    }  
}
```

```
@Override  
    public boolean equals(Object o) {  
        if (this == o) return true;  
        if (o == null || getClass() !=  
o.getClass()) return false;  
  
        Person person = (Person) o;  
  
        if (name != null ?  
!name.equals(person.name) : person.name != null)  
return false;  
        if (age != 0 ? age != person.age :  
person.age != 0) return false;  
    }  
  
    @Override  
    public int hashCode() {  
        int result = name != null ? name.hashCode()  
: 0;  
        result = 31 * result + age;  
        return result;  
    }  
    @Override  
    public String toString() {  
        return "Person{" +  
            "name='" + name + '\'' +  
            ", age='" + age + '\'' +  
            '}';  
    }  
    Person person = new Person("John", 34);  
}
```

Kotlin - classes

```
data class Person(var name: String, var age: Int)  
  
val person : Person = Person("John", 34)
```

The compiler automatically derives the following members from all properties declared in the primary constructor:

- `equals()/hashCode() pair`
- `toString()` of the form "User(name=John, age=42)"
- `componentN() functions` corresponding to the properties in their order of declaration.
- `copy() function.`

Java - cast

```
public void demo(Object x) {  
    if (x instanceof String) {  
        System.out.println(((String)x).length;  
    }  
}
```

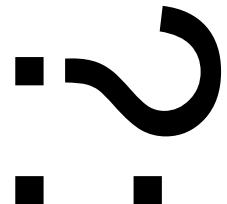
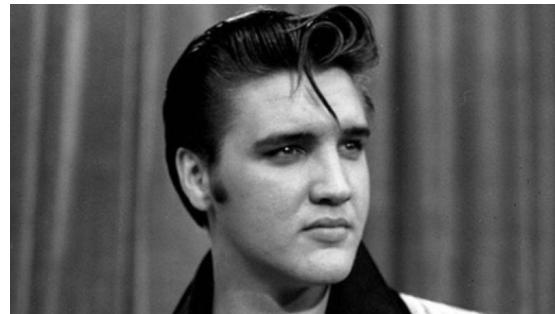
Kotlin - cast

```
fun demo(x: Any) {  
    if (x is String) {  
        println(x.length) //x is automatically casted to String  
    }  
}
```

Link: <https://kotlinlang.org/docs/typecasts.html#what-kotlin-has-that-java-does-not>

Java - null

```
public int length(String word) {  
    int length = 0;  
    if (word != null) {  
        length = word.length;  
    } else {  
        length = 0;  
    }  
    return length;  
}
```

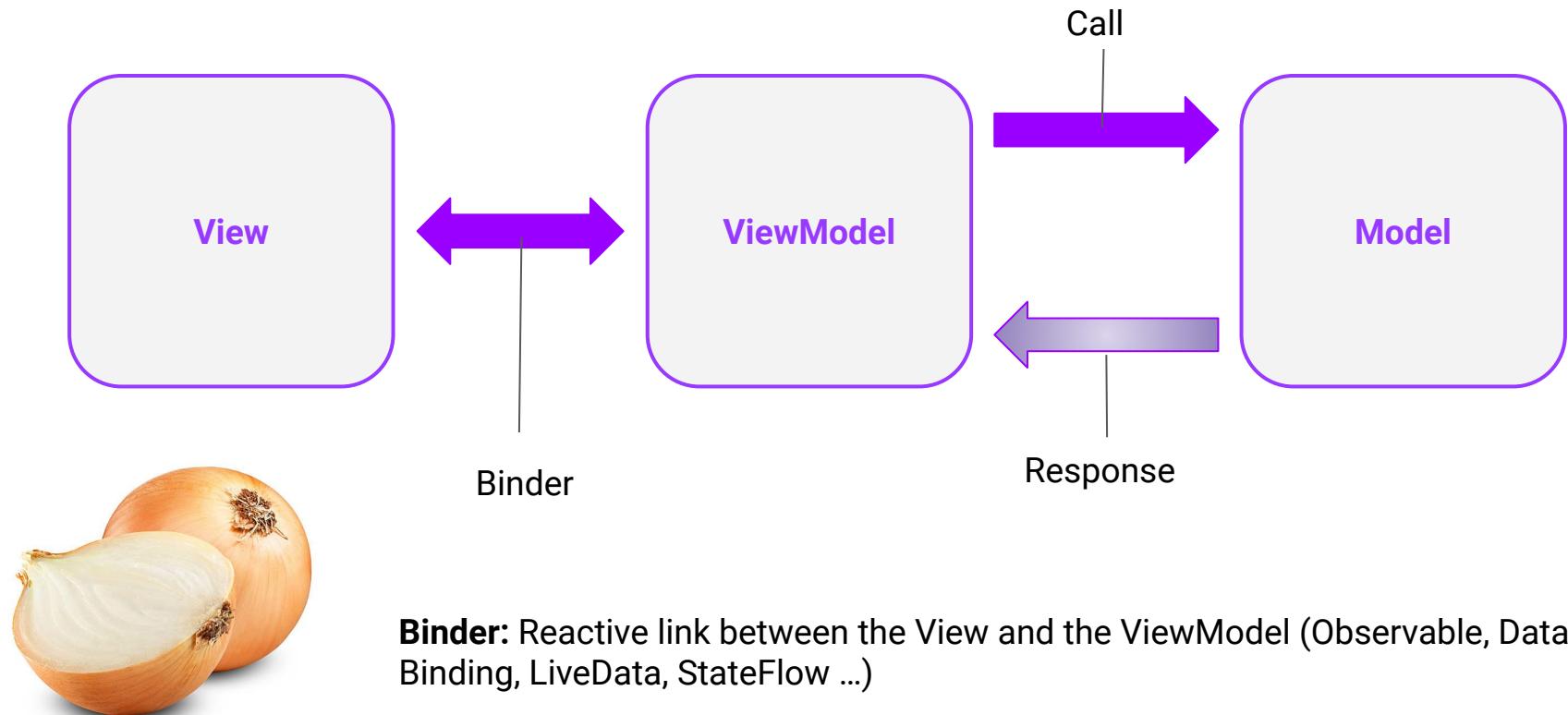


Kotlin - null

```
fun length(word: String?) = word?.length ?: 0
```

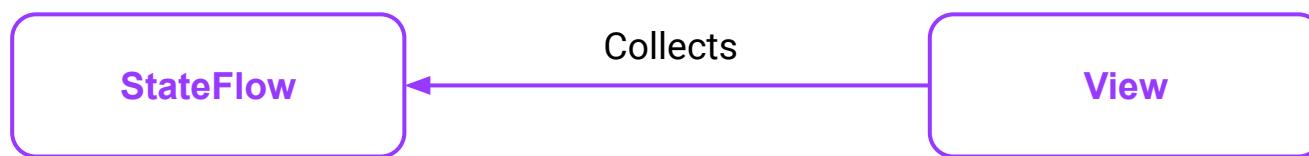
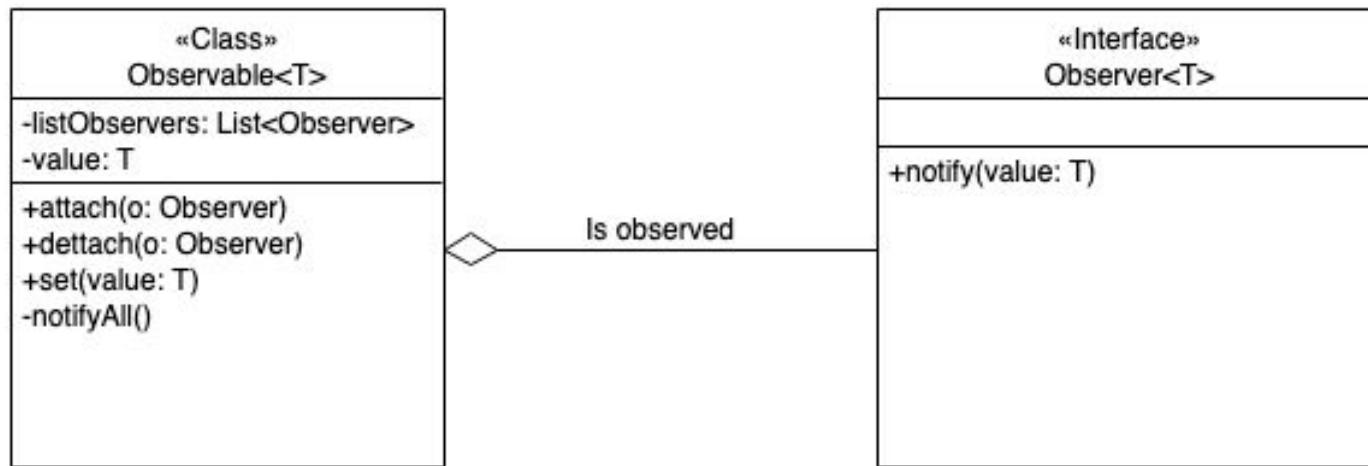
Link: <https://kotlinlang.org/docs/null-safety.html>

MVVM: Architecture



Android: development

MVVM: Observer



Link Kotlin Flow: <https://kotlinlang.org/docs/flow.html>

Android: development

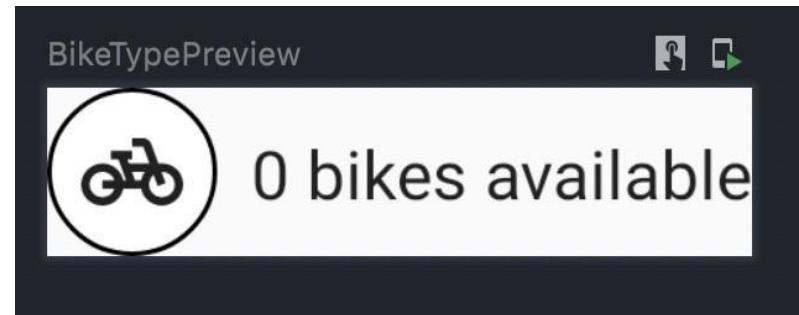
Compose

- Creation of Graphical User Interfaces (GUI)
- Declarative
- Similar to React, Flutter, SwiftUI...
- Compatible with Android classic Views



Link Compose tutorial: <https://developer.android.com/jetpack/compose/tutorial>

```
@Composable
private fun BikeDetail(
    modifier: Modifier = Modifier,
    icon: ImageVector,
    text: String,
) {
    Row(
        modifier = modifier,
        verticalAlignment = Alignment.CenterVertically,
    ) {
        Surface(
            shape = CircleShape,
            border = BorderStroke(1.dp, MaterialTheme.colors.onSurface),
        ) {
            Icon(
                imageVector = icon,
                contentDescription = null,
                modifier = Modifier.padding(8.dp),
            )
        }
        Text(
            text = text,
            style = MaterialTheme.typography.body1,
            modifier = Modifier.padding(start = 8.dp),
        )
    }
}
```



Methodology and Tools

Git

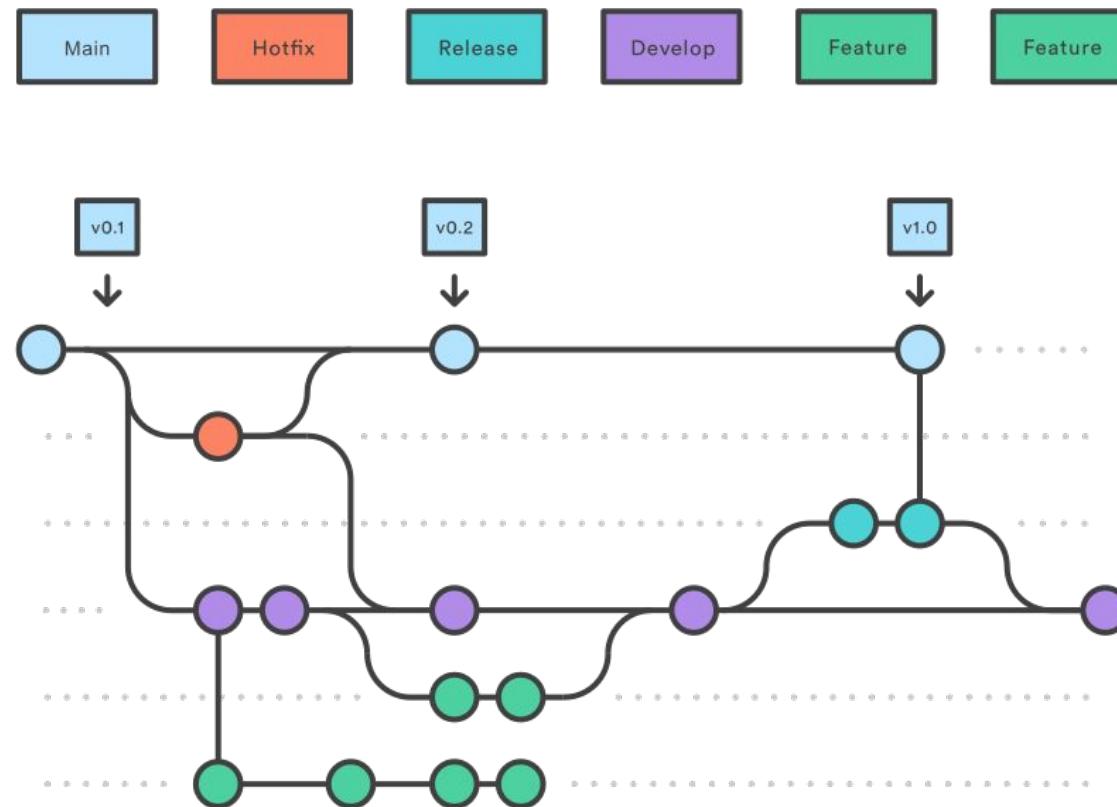
- Distributed version control tool
- Free and open, designed by Linus Torvalds



Git tutorial: <https://www.w3schools.com/git/>

Git GUIs: <https://git-scm.com/downloads/guis/>

Gitflow



Link: <https://www.atlassian.com/en/git/tutorials/comparing-workflows/gitflow-workflow>

Linters

A linter is a static code analyser that automatically checks the application source code for programmatic and stylistic errors.



Link: <https://www.perforce.com/blog/qac/what-lint-code-and-why-linting-important>

CI/CD

Continuous Integration (CI) is the process of automation in the source code integration from various contributors in the same software project.

Continuous Delivery (CD) is the action of launching automation test processes before uploading the code of a developer into a repository.

Continuous Deployment (CD) is the process of deploying the changes from the repository to production automatically.

Links:

<https://www.atlassian.com/es/continuous-delivery/continuous-integration>
<https://www.redhat.com/es/topics/devops/what-is-ci-cd>

Software



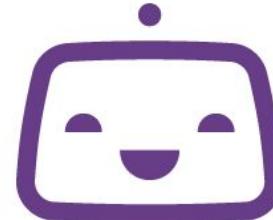
Jenkins



Azure DevOps



GitHub Actions



And many more...

Jenkins + Gitlab

Stage View



Pipeline #13518 passed for 7a6ab2f4 on AMBU-XX_some_tablet_issues 1 month ago



Jenkins + Gitlab

Refactor location availability.

⌚ 6 jobs for [AMBU-XX_change_gps_provider](#) in 0 seconds (queued for 2 seconds)

⟳ latest

-o- [ac945958](#) ↗

💡 1 related merge request: [!178 AMBU-XX: Change GPS provider](#)

Pipeline Needs Jobs 6 Tests 0

External

✓ 1 - checkout

✓ 2 - build

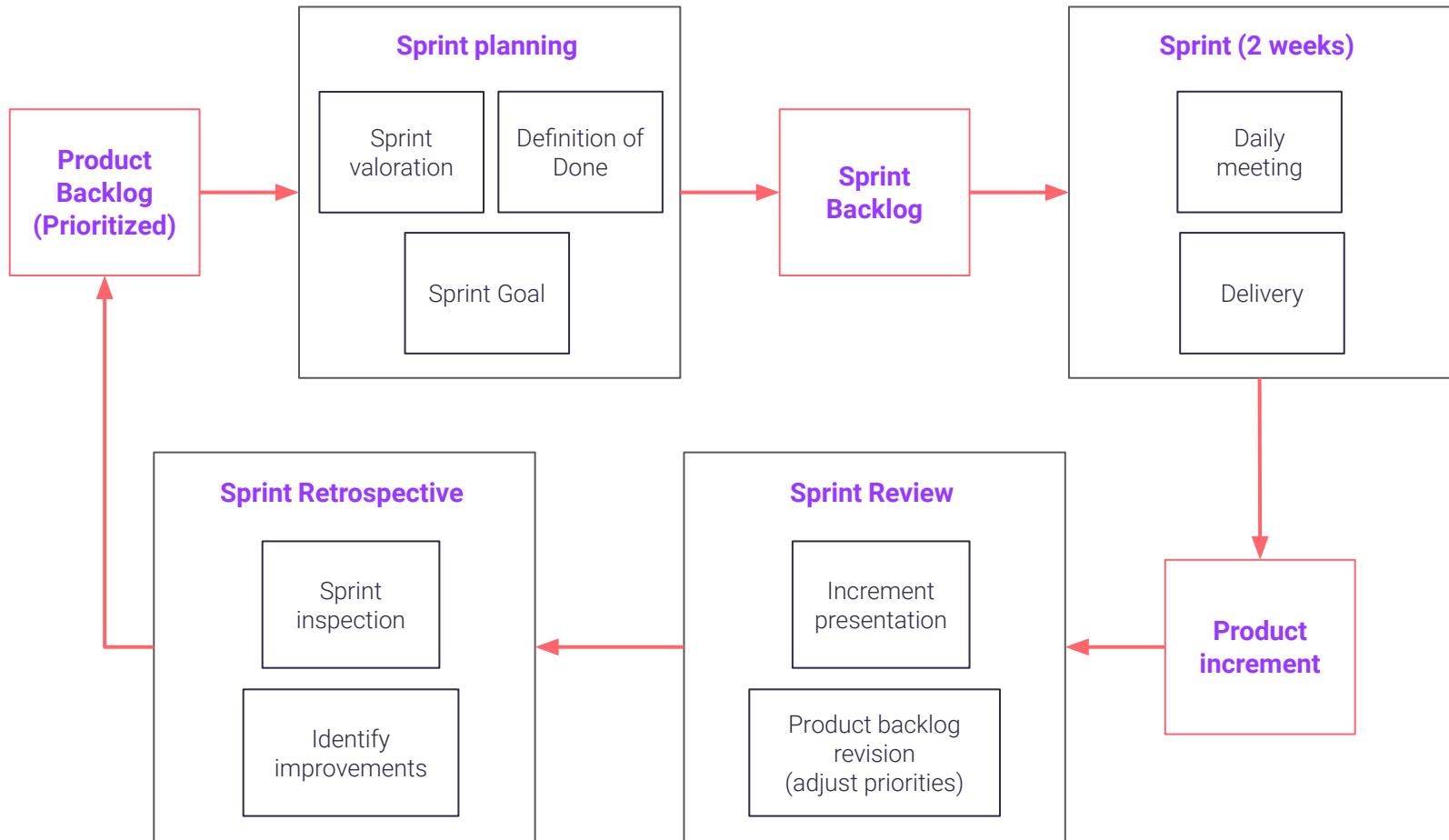
✓ 3 - ktlint

✓ 4 - lint

✓ 5 - detekt

✗ 6 - Unit test

Agile



Methodology and Tools

Jira

Proyectos / Basetis App RELOAD / Epic Backlog

Tablero de Kanban

Buscar en este tablero   Solo Mis Incidencias Recientemente Actualizadas

OPEN 9

- Clock in-out
 - Clock in-out
 - Ninguna
 -  BA-866
- Agrupa funcionalidades de envío, recepción y gestión de notificaciones
 - Notifications
 - Ninguna
 -  BA-114

TO REFINEMENT 5

- Añadir gráfico a información salarial del empleado
 - Añadir gráfico a información sal...
 - Ninguna
 -  BA-853
- Pay roll information including bonus
 - Pay roll information including b...
 - Ninguna
 -  BA-868

READY FOR DEVELOPMENT 4

- My payrolls
 - My payrolls
 - Ninguna
 -  BA-814
- Información salarial del empleado - sin gráfico
 - Información salarial del emplea...
 - v3.2.0
 -  BA-803
- Flexible Retribution section

BLOCKED 0

IN PROGRESS 2

- Register Redesign
 - Register Redesign
 - v3.2.0
 -  BA-812
- Absence request
 - Absence request
 - v3.1.0
 -  BA-838

QA

Resources

- Kotlin koans: <https://play.kotlinlang.org/koans/overview>
- Android Studio: <https://developer.android.com/studio>
- Android with Compose: <https://developer.android.com/courses/android-basics-compose/course>
- MVVM: <https://developer.android.com/topic/libraries/architecture/viewmodel>
- REST requests: <https://square.github.io/retrofit/>
- Clean Architecture:
<https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>

Contact

- albert.pinto@basetis.com
- iris.serrano@basetis.com
- julian.zaragoza@basetis.com

Live coding!

Repository: <https://git.basetis.com/basetis-public/BicingAround>

basetis

jordi.saumell@basetis.com
www.basetis.com

Segueix-nos a @Basetis



2022