{EPITECH.}
PARIS GRADUATE SCHOOL OF DIGITAL INNOVATION

Cash Manager

Kick-off

T7 - Application Development

T-DEV-700

Dev App

In professional context, developing software is much more than writing code.

Dev App

In professional context, developing software is much more than writing code.

Many additional challenges come with working as a team, on big projects, with the aim to satisfy clients.

Challenges

- **Specifications**: understanding clients explicit/implicit demands and turning them into formal requirements

Challenges

- **Specifications**: understanding clients explicit/implicit demands and turning them into formal requirements
- **Modularity**: organizing a project into manageable, independent functionalities

Challenges

- **Specifications**: understanding clients explicit/implicit demands and turning them into formal requirements
- **Modularity**: organizing a project into manageable, independent functionalities
- **Robustness**: building code that remains efficient with context variation and technical evolution

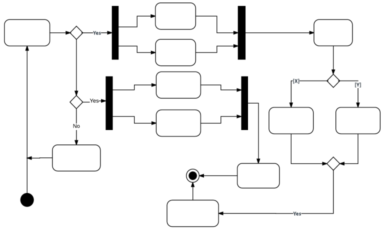
Challenges

- **Specifications**: understanding clients explicit/implicit demands and turning them into formal requirements
- **Modularity**: organizing a project into manageable, independent functionalities
- **Robustness**: building code that remains efficient with context variation and technical evolution
- **Documentation**: writing readable code and documentation for easy use and further third-party development

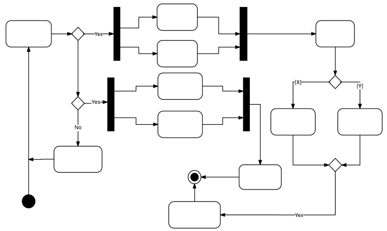
Challenges

- **Specifications**: understanding clients explicit/implicit demands and turning them into formal requirements
- **Modularity**: organizing a project into manageable, independent functionalities
- **Robustness**: building code that remains efficient with context variation and technical evolution
- **Documentation**: writing readable code and documentation for easy use and further third-party development
- **Efficiency**: avoiding unnecessary work and making cooperation easier through the use of well-known patterns and tools

Before and around coding

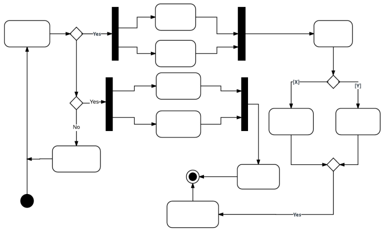


Before and around coding



For big projects, it is essential to divide the work charge into manageable entities.

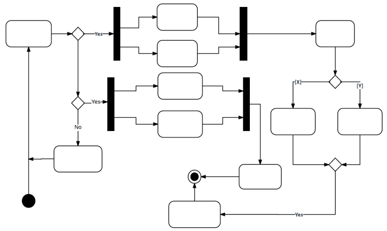
Before and around coding



For big projects, it is essential to divide the work charge into manageable entities.

Formal specifications lead to user stories, code organization, activity sequences,. . .

Before and around coding



For big projects, it is essential to divide the work charge into manageable entities.

Formal specifications lead to user stories, code organization, activity sequences,. . .

At the end of this process, several teams have clear roadmaps that allow them to work autonomously.

Tools

JAVA eclipse web service SWING! JSP

EJB 3 JavaFx spring HIBERNATE

LOG4J JDBC PRIME FACES jQuery write less, do more.

JUnit Struts2 Apache Tomcat XML

Java Servlets Java EE maven {JSON}

Tools

JAVA
eclipse
web service
SWING!
JSP
EJB 3
JavaFx
spring
HIBERNATE
LOG4J
JDBC
PRIME FACES
jQuery write less, do more.
JUnit
Struts2
Apache Tomcat
XML
Java Servlets
Java EE
maven
{JSON}

A good work environment comes with a whole ecosystem for handling dependencies, avoiding boilerplate code, sharing files, testing code, or deploying product.

Tools

JAVA
eclipse
web service
SWING!
JSP
EJB 3
JavaFx
spring
HIBERNATE
LOG4J
JDBC
PRIMEFACES
jQuery
JUnit
Struts2
Apache Tomcat
XML
Java Servlets
Java EE
maven
{JSON}

A good work environment comes with a whole ecosystem for handling dependencies, avoiding boilerplate code, sharing files, testing code, or deploying product.

Some are specific to a language while others may be more generalists.

OOP and Java

- decompose in small, ordered entities

OOP and Java

- decompose in small, ordered entities
- highly constrained and verbose syntax

OOP and Java

- decompose in small, ordered entities
- highly constrained and verbose syntax
- easy to understand and reproduce

OOP and Java

- decompose in small, ordered entities
- highly constrained and verbose syntax
- easy to understand and reproduce
- limited surprises at execution

OOP and Java

- decompose in small, ordered entities
- highly constrained and verbose syntax
- easy to understand and reproduce
- limited surprises at execution
- power of the JVM

The not-only-Internet of Things

The not-only-Internet of Things

Many real-life applications are not mere pieces of code running on your laptop, but include being able to handle the physical behavior of a device.

The not-only-Internet of Things

Many real-life applications are not mere pieces of code running on your laptop, but include being able to handle the physical behavior of a device.

Here you will be expected to deal with mainstream features: **transmission**, **camera**, **scanning**.

CashManager

You will build two different entities:

CashManager

You will build two different entities:

- a **kotlin android application** that is able to scan articles and recognize credit card for payment

CashManager

You will build two different entities:

- a **kotlin android application** that is able to scan articles and recognize credit card for payment
- a **java server** that communicates with the app, issues billing operations and produces responses.

CashManager

You will build two different entities:

- a **kotlin android application** that is able to scan articles and recognize credit card for payment
- a **java server** that communicates with the app, issues billing operations and produces responses.

Beyond the development of the app, this project is a first (big) step in handling the whole production ecosystem.

CashManager

Developping robust and re-usable code is not an option!

CashManager

Developping robust and re-usable code is not an option!

- Design Pattern

CashManager

Developping robust and re-usable code is not an option!

- Design Pattern
- Code Coverage

CashManager

Developping robust and re-usable code is not an option!

- Design Pattern
- Code Coverage
- Documentation

CashManager

Developping robust and re-usable code is not an option!

- Design Pattern
- Code Coverage
- Documentation
- Maven, Docker

Any questions

?