

# Agilité et **Gitlab**

J.-M. Bruel

Version 1.0 2019-01-12

# Table des matières

1. 🐾 Les clients de votre projet . . . . .	1
2. 📖 Cahier Des Charges Utilisateur . . . . .	2
2.1. Késako . . . . .	2
2.2. [fa yin yang] Extra-functional properties . . . . .	2
2.3. Design . . . . .	3
2.4. Autre exemple . . . . .	3
3. [mail bulk] Pour votre <b>Projet</b> . . . . .	3
3.1. 🎯 Focus . . . . .	3
3.2. 🔄 Correspondances de concepts Scrum/Gitlab . . . . .	4
3.3. ⚙️ Build . . . . .	4
3.4. git . . . . .	7
3.5. git Bonnes pratiques . . . . .	7
4. 🐾 <b>Exemple</b> : génération de la doc . . . . .	7

# 1. 👥 Les clients de votre projet

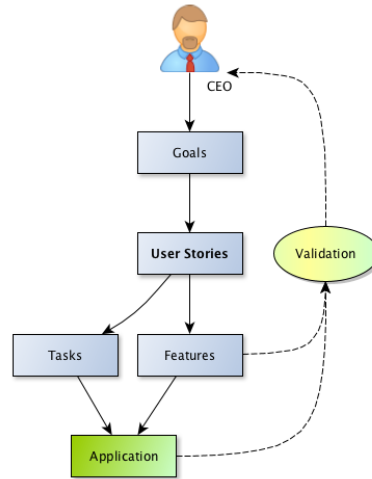


Figure 1. Le client (qui va utiliser le logiciel)

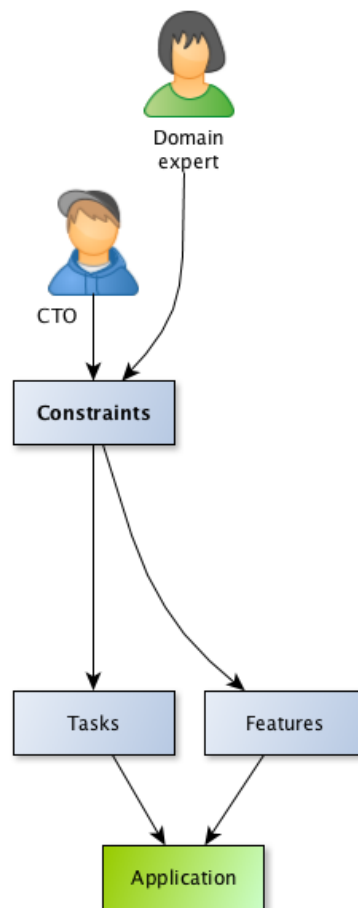


Figure 2. Le prof (qui va contraindre le logiciel)

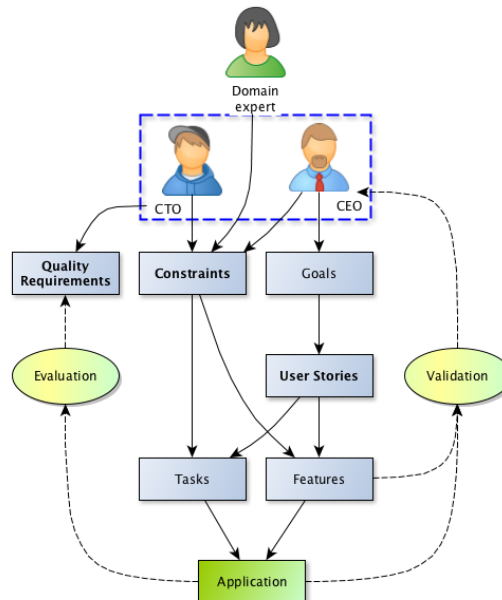


Figure 3. Le prof (qui va noter votre projet)



## 2. Cahier Des Charges Utilisateur

### 2.1. Késako

- "Contrat" entre le client et vous
- [QQOQCCP](#)
- Qui est responsable de quoi (Client/MOA/MOE) ?
- Contexte
- Contraintes
- Liste des fonctionnalités attendues
- Cf. [Moodle](#)

### 2.2. [fa yin yang] Extra-functional properties

Extra-functional properties are sometimes more important than functional ones...

	 Cell Phones Before & now 	
Battery Life:	3-4 Days.	3-4 hours.
Bump limit without breaking:	3 Floor (asphalt)	Half a meter (parquet)
Bumps protection:	Integrated.	Purchased separately.
Software updates:	Not needed.	Approx. once a week.
Approx. Lifetime:	∞ (Still works)	Max. 1-2 years.
Typing speed when writing a text:	9 characters per second.	Damn you autocorrect!

## 2.3. Design

Quelques exemples de propriétés extra fonctionnelles...



## 2.4. Autre exemple



© Dinga

## 3. [mail bulk] Pour votre Projet

### 3.1. 📌 Focus

-  [GitLab](#)
- build
- git

## 3.2. ⚡ Correspondances de concepts Scrum/Gitlab

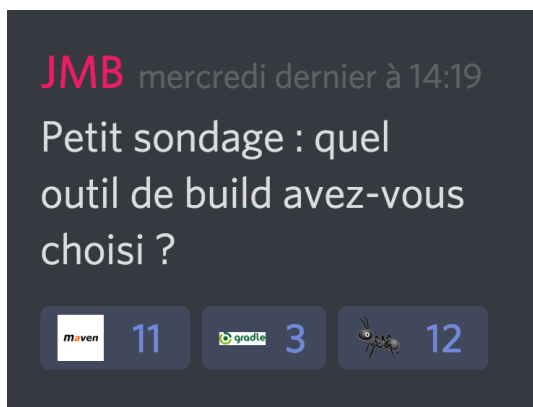
### Mapping Agile artifacts to GitLab features

Agile artifact	GitLab feature
User story	<a href="#">Issues</a>
Task	<a href="#">Task lists</a>
Epic	<a href="#">Epics</a>
Points and estimation	<a href="#">Weights</a>
Product backlog	<a href="#">Issue lists and prioritized labels</a>
Sprint/iteration	<a href="#">Milestones</a>
Burndown chart	<a href="#">Burndown charts</a>
Agile board	<a href="#">Issue boards</a>

Figure 4. Correspondances entre Artefacts

## 3.3. ⚙️ Build

Résultats du sondage Discord :



Etude 2014 :

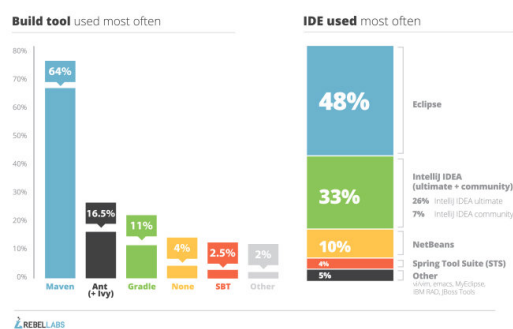


Figure 5. Ant vs Maven vs Gradle (source [here](#))

```
<project xmlns:ivy="antlib:org.apache.ivy.ant" name="java-build-tools" default="jar">

  <property name="src.dir" value="src"/>
  ...
  <path id="lib.path.id">
    <fileset dir="${lib.dir}" />
  </path>

  <target name="clean">
    <delete dir="${build.dir}"/>
  </target>

  <target name="compile">
    <mkdir dir="${classes.dir}"/>
    <javac srcdir="${src.dir}" destdir="${classes.dir}"
classpathref="lib.path.id"/>
  </target>

  <target name="jar" depends="compile">
    <mkdir dir="${jar.dir}"/>
    <jar destfile="${jar.dir}/${ant.project.name}.jar" basedir="${classes.dir}"/>
  </target>

</project>
```

Exemple de fichier Maven (cf. [source](#))

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">

  <modelVersion>4.0.0</modelVersion>
  <groupId>com.technologyconversations</groupId>
  <artifactId>java-build-tools</artifactId>
  <packaging>jar</packaging>
  <version>1.0</version>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>4.11</version>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>2.3.2</version>
      </plugin>
    </plugins>
  </build>

</project>
```

Exemple de fichier Gradle (cf. [source](#))

```
apply plugin: 'java'
apply plugin: 'checkstyle'
apply plugin: 'findbugs'
apply plugin: 'pmd'

version = '1.0'

repositories {
  mavenCentral()
}

dependencies {
  testCompile group: 'junit', name: 'junit', version: '4.11'
  testCompile group: 'org.hamcrest', name: 'hamcrest-all', version: '1.3'
}
```

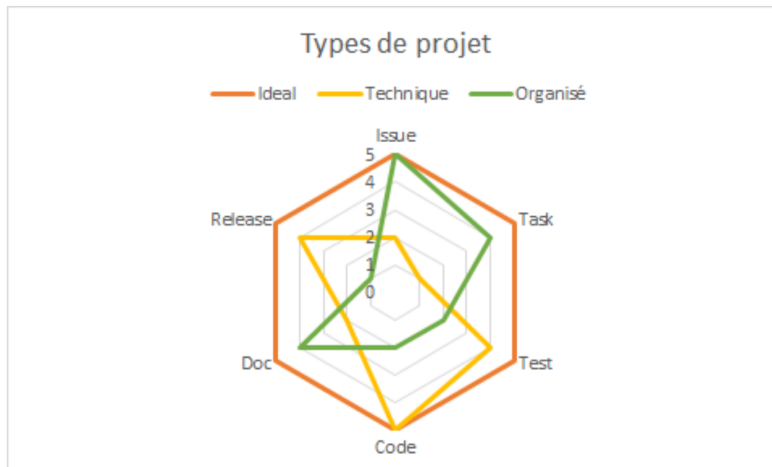


### 3.4. git

cf. <http://bit.ly/jmb-git>

### 3.5. git Bonnes pratiques

<https://github.com/xblanc33/QualiteDev>



## 4. 👤 Exemple : génération de la doc

```
image: asciidoctor/docker-asciidoctor
```

```
variables:
```

```
  GIT_SSL_NO_VERIFY: "1"
```

```
stages:
```

- build
- test

```
html:
```

```
  stage: build
```

```
  script:
```

- asciidoctor README.adoc -o index.html

```
artifacts:
```

- paths:
- index.html

```
pdf_preview:
```

```
  stage: test
```

```
  when: manual
```

```
  environment:
```

```
    name: preview/${CI_COMMIT_REF_NAME}
```

```
  except:
```

- /master/

```
artifacts:
```

- paths:
- README.pdf
- expire\_in: 1 week

```
script:
```

- asciidoctor-pdf README.adoc

