



DE L'INTEGRATION AU DEPLOIEMENT CONTINU

L'automatisation de la livraison logiciel au
cœur du mouvement DEVOPS

CONFERENCE TECHNIQUE 2017

LUNDI 22 MAI 2017
CESi LABEGE

- KEYNE VIEIRA
- OLIVIER CHABRIER
- JOHANN COTTEVERTE
- ROMAIN JANSSEN



CESi
alternance

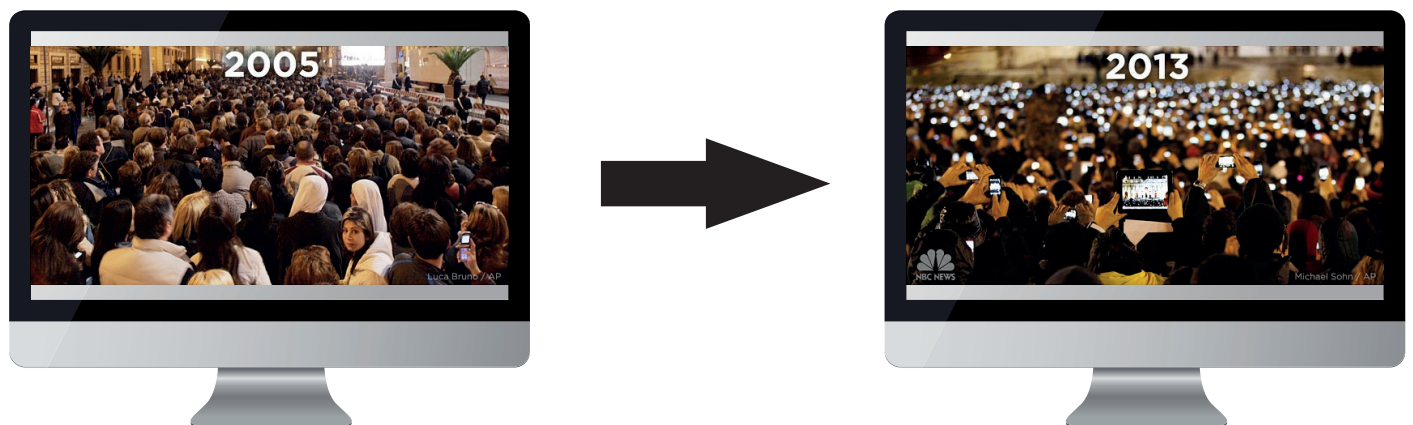
ÉCOLE SUPÉRIEURE DES MÉTIERS

Les évolutions du numérique

Transformation numérique

Le paysage numérique a complètement changé en l'espace de quelques années seulement. En effet, lorsque nous regardons les chiffres officiels du ministère de l'économie, le pourcentage de personnes ayant un smartphone est passé de 29% en 2012 à 58% en 2015. Nous constatons la même chose pour les tablettes qui ont bondi de 8% en 2012 à 35% en 2015.

Cette augmentation très forte de possession de terminaux mobiles numériques a pour conséquence une explosion de la consommation des services numériques (Web, Applications, Messagerie, Jeux, ...).



Exemple marquant de l'impact cette transformation numérique

Évolution des besoins

Le monde a évolué, et les besoins aussi. Sous l'impulsion de l'arrivée de nouvelles générations, d'abord la génération Y (née entre 1980-1995) puis la génération Z (>1995), les marques doivent sans cesse se transformer, évoluer pour répondre aux nouveaux codes et modes de consommation de leurs clients.

Les évolutions des consommations ont bien sûr été portées par l'émergence des nouvelles technologies et du monde connecté mais pas uniquement. Les consommateurs ont aussi évolué plus structurellement en termes de besoins et d'attentes. Nous avons un besoin de personnaliser nos produits. Alors que dans les années 90, nos téléphones avaient des fonctionnalités basiques, aujourd'hui le consommateur veut un produit qui lui ressemble. Nous nous différencions et il faut répondre à ces besoins.



Internet, les applications et les smartphones ont définitivement changé notre vie.

Les évolutions du numérique

Évolution des applications

En quelques années, l'évolution des applications a totalement bouleversé notre vie. Dans cette mouvance, les habitudes des utilisateurs ont aussi évolué.

De l'internet en passant par les smartphones, notre façon de consommer les applications a été modifiée, a augmenté au fil du temps et nous demandons toujours plus de fonctionnalités !



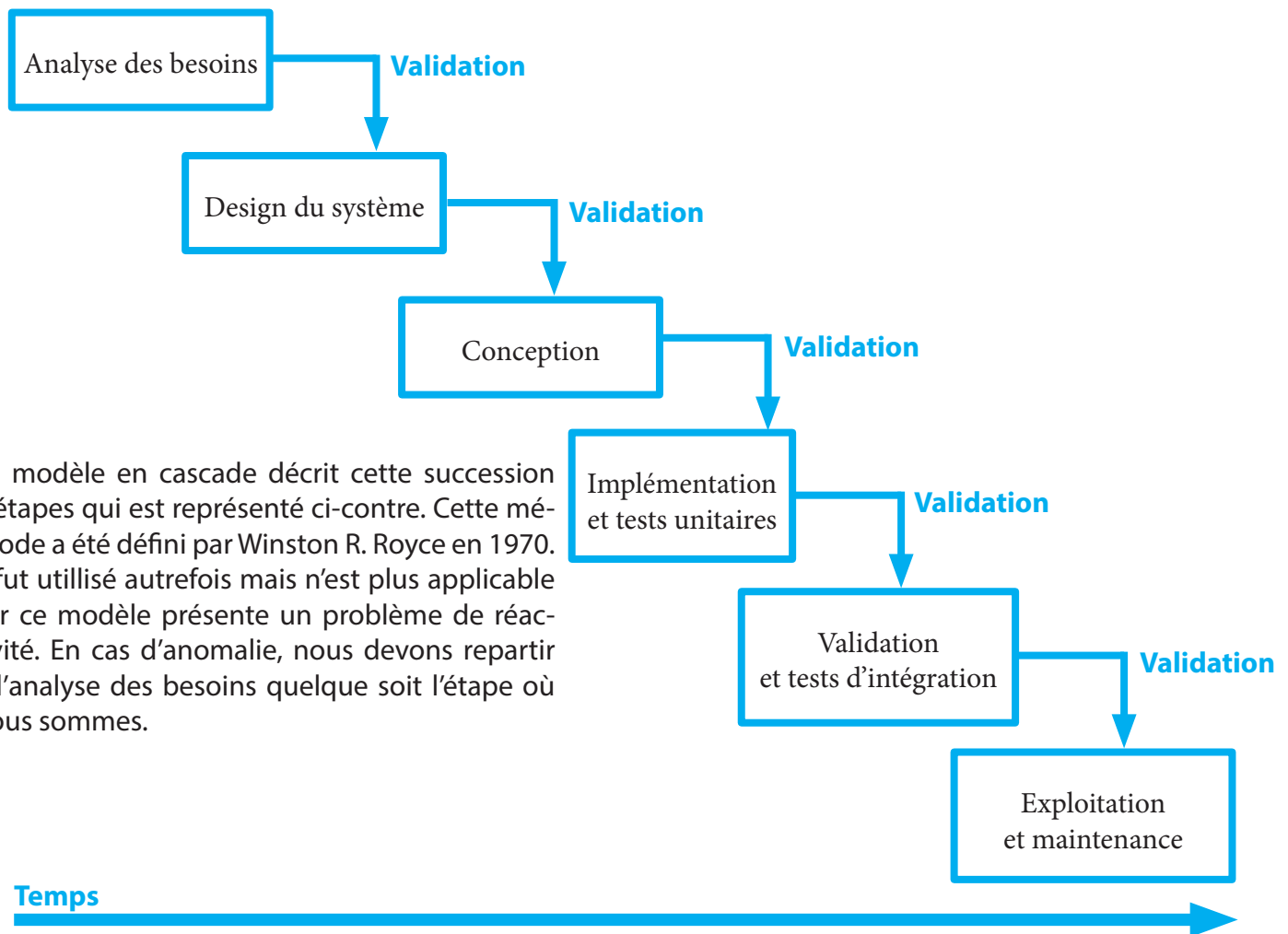
Les évolutions du numérique

Évolution des méthodes

Afin de répondre à des exigences de plus en plus diverses et variées, les méthodes de travail ont dû s'adapter et évoluer.

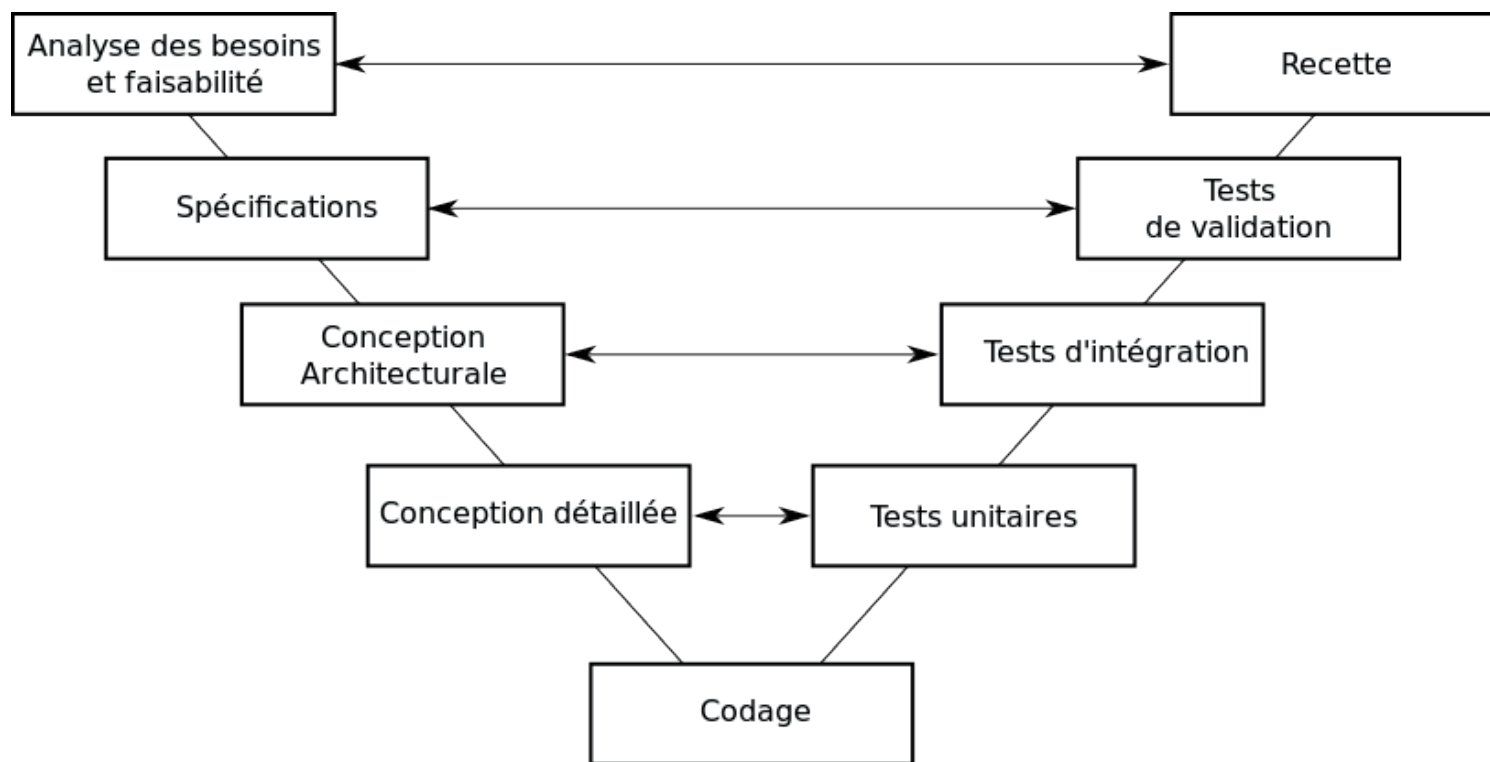
La majorité des entreprises du numérique sont « agiles » dans le but de pouvoir répondre et satisfaire leurs clients au plus vite.

Waterfall (modèle en cascade)



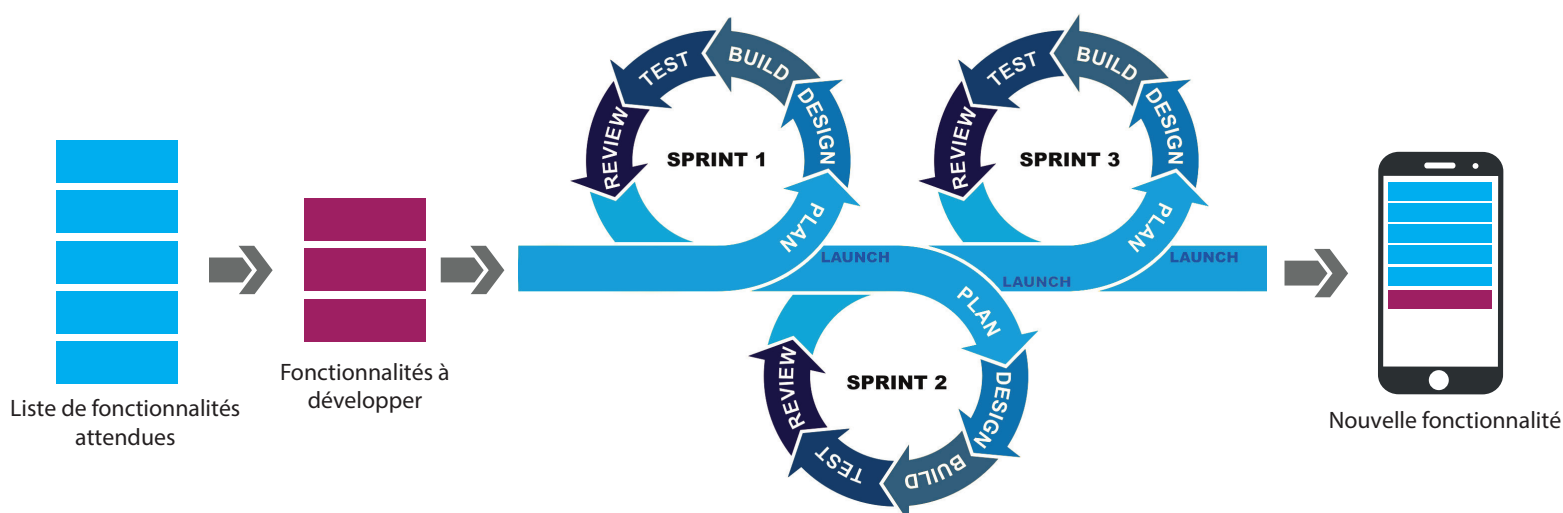
Les évolutions du numérique

Cycle en V



Le cycle en V, plus réactif que le modèle en cascade, permet de limiter un retour aux étapes précédentes en cas d'anomalie.

Méthode Agile



Les méthodes de développement dites « méthodes agiles » visent à réduire le cycle de vie du logiciel (donc accélérer son développement) en développant une première version, puis en intégrant les fonctionnalités par un processus itératif basé sur une écoute client et des tests tout au long du cycle de développement.

Les grandes étapes du code

Cette étape représente la phase de production du code source.

Elle inclut :

- la rédaction
- le stockage dans un dépôt de code source
- la gestion de la configuration

CODE

L'intégration est un ensemble de pratiques utilisées en génie logiciel consistant à vérifier à chaque modification de code source que le résultat des modifications ne produit pas de régression dans l'application développée.

Le concept a pour la première fois été mentionné par Grady Booch.

Le principal but de cette pratique est de détecter les problèmes d'intégration au plus tôt lors du développement.

INTEGRATION

En informatique, la génération désigne le processus de transformation d'un programme écrit dans un langage lisible par un humain en un programme exécutable par une machine.

De manière plus générale, il s'agit de traduire un programme écrit dans un langage source en un programme écrit dans un langage cible.

GENERATION

À la mise en production

Etape qui consiste à déployer l'application obtenue à la phase précédente sur un serveur dédié aux tests pour effectuer la recette.

DEPLOIEMENT SERVEUR
DE
TESTS

L'application est soumise à des tests de qualification (recette) sur un serveur de tests afin que le responsable qualité puisse vérifier la conformité de l'application aux besoins.

Pour cela, il s'appuie sur un plan de tests rédigé en amont. Ce plan de tests contient la liste des actions à tester et les résultats attendus.

L'ensemble des tests sont réalisés sur l'application, et le responsable qualité s'assure que les résultats effectifs soient équivalents aux fonctionnalités attendues.

Si ces tests s'avèrent concluants, on passe à la mise en production.

RECETTE

Etape finale qui consiste à déployer l'application sur le serveur de production dans le but de la rendre disponible à l'ensemble des utilisateurs.

MISE
EN
PRODUCTION

Quelques outils de l'automatisation



VS code :

Visual Studio Code est un environnement de développement cross-platform, open source et gratuit, supportant une dizaine de langages.

En programmation informatique, un environnement de développement est un ensemble d'outils pour augmenter la productivité des programmeurs qui développent des logiciels. Il comporte un éditeur de texte destiné à la programmation, des fonctions qui permettent, par pression sur un bouton, de démarrer le compilateur ou l'éditeur de liens ainsi qu'un débogueur en ligne, qui permet d'exécuter ligne par ligne le programme en cours de construction.



GitHub

Github :

GitHub est un service web d'hébergement et de gestion de code source, utilisant le logiciel de gestion de versions Git. GitHub propose des comptes professionnels payants, ainsi que des comptes gratuits pour les projets de logiciels libres.

Un logiciel de gestion de versions (ou VCS en anglais, pour version control system) est un logiciel qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus. Il permet notamment de retrouver les différentes versions d'un lot de fichiers connexes.

Github est ce que l'on appelle un dépôt, en informatique, un dépôt ou référentiel (de l'anglais repository) est un stockage centralisé et organisé de données.

Quelques outils de l'automatisation



CODESHIP

Codeship :

Codeship est un serveur d'intégration continue créé en 2010 par Moritz Plassnig, Manuel Weiss, and Florian Motlik.

L'intégration continue est un ensemble de pratiques utilisées en génie logiciel consistant à vérifier à chaque modification de code source que le résultat des modifications ne produit pas de régression dans l'application développée. Le concept a pour la première fois été mentionné par Grady Booch.

Le principal but de cette pratique est de détecter les problèmes d'intégration au plus tôt lors du développement. De plus, elle permet d'automatiser l'exécution des suites de tests et de voir l'évolution du développement du logiciel.

Les serveurs d'intégration continue sont de plus en plus utilisés en entreprise afin d'améliorer la qualité du code et du produit final.



simple, flexible, fun

Mocha :

Mocha est un framework Javascript permettant de réaliser des tests sur ses développements JavaScript. Il permet de réaliser des tests unitaires, et des tests d'intégration.

Dans le monde du développement informatique, le test d'intégration est la phase qui est précédée des tests unitaires et est généralement suivie par les tests de validation. Dans le test unitaire, on vérifie le bon fonctionnement d'une partie précise d'un logiciel ou d'une portion d'un programme (appelée « unité » ou « module ») ; dans le test d'intégration, chacun des modules indépendants du logiciel est assemblé et testé dans l'ensemble.



webpack

Webpack :

Webpack est ce que l'on appelle un "module bundler". Comme son nom l'indique, il permet dans le cadre du développement web de packager le code source.

Pour cela, il va fusionner l'ensemble des fichiers de code en un fichier unique, facilitant le déploiement.

Cette pratique en JavaScript est l'équivalent de la compilation dans d'autre langage tel que Java ou C#.

Quelques outils de l'automatisation



Nightwatch :

Nightwatch est un framework de test end to end (de bout en bout).

Le test de bout en bout est une méthodologie utilisée pour tester si une application fonctionne comme prévu du début à la fin.

Nightwatch permet de réellement simuler les actions des utilisateurs sur le site ou l'application web. Il est donc possible d'exécuter les plans de test de validation de manière automatique. De plus, ces tests pourront être effectués sur de multiples navigateurs (IE, Chrome, Firefox, etc.).



Excel :

Microsoft Excel est un logiciel tableur de la suite bureautique Microsoft Office, développée et distribuée par l'éditeur Microsoft.

Il permet dans notre cas de figure de rédiger les plans de tests manuscrits sous forme de tableau. Ce tableau contient une liste d'actions à réaliser sur l'application associée à un comportement attendu.

Lorsque le testeur joue les plans de tests, il effectue les actions et vérifie que le comportement de l'application soit équivalent au comportement prévu.



Heroku :

Heroku est un service de cloud computing de type plate-forme en tant que service. Créé en 2007, il était l'un des tout premiers services cloud.

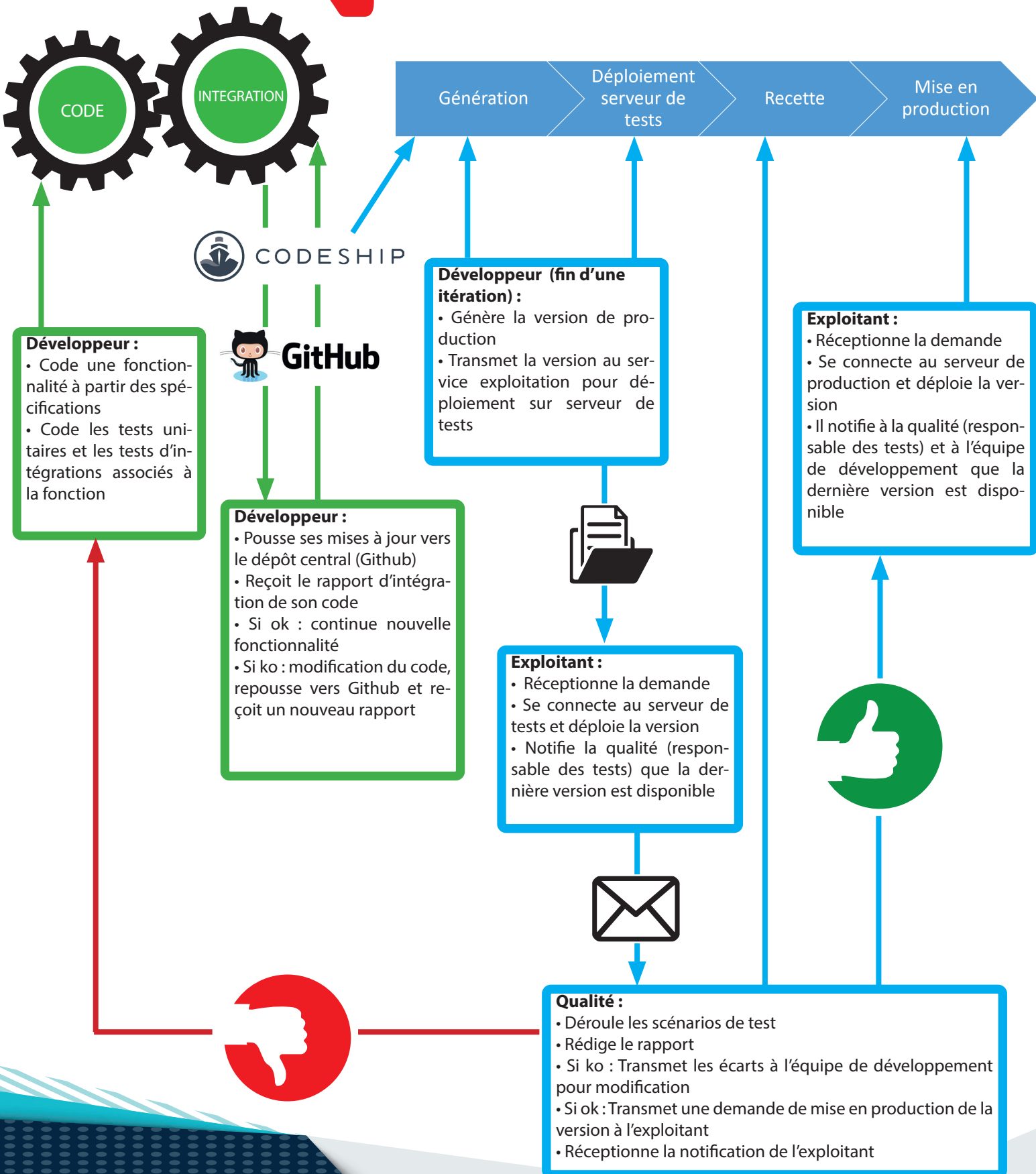
Le service permet le déploiement très rapide d'applications web dans le cloud avec une gestion très souple.

L'infrastructure utilisée par Heroku repose sur AWS (Amazon Web Services). Différents services d'AWS, notamment EC2 (pour l'exécution du code) et S3 (pour le stockage) sont mis à profit dans l'architecture Heroku pour mettre à disposition de l'utilisateur final un environnement de déploiement et d'exécution adapté à son besoin.

Intégration continue

Automatisation

Étapes non automatique



Le saviez-vous ?

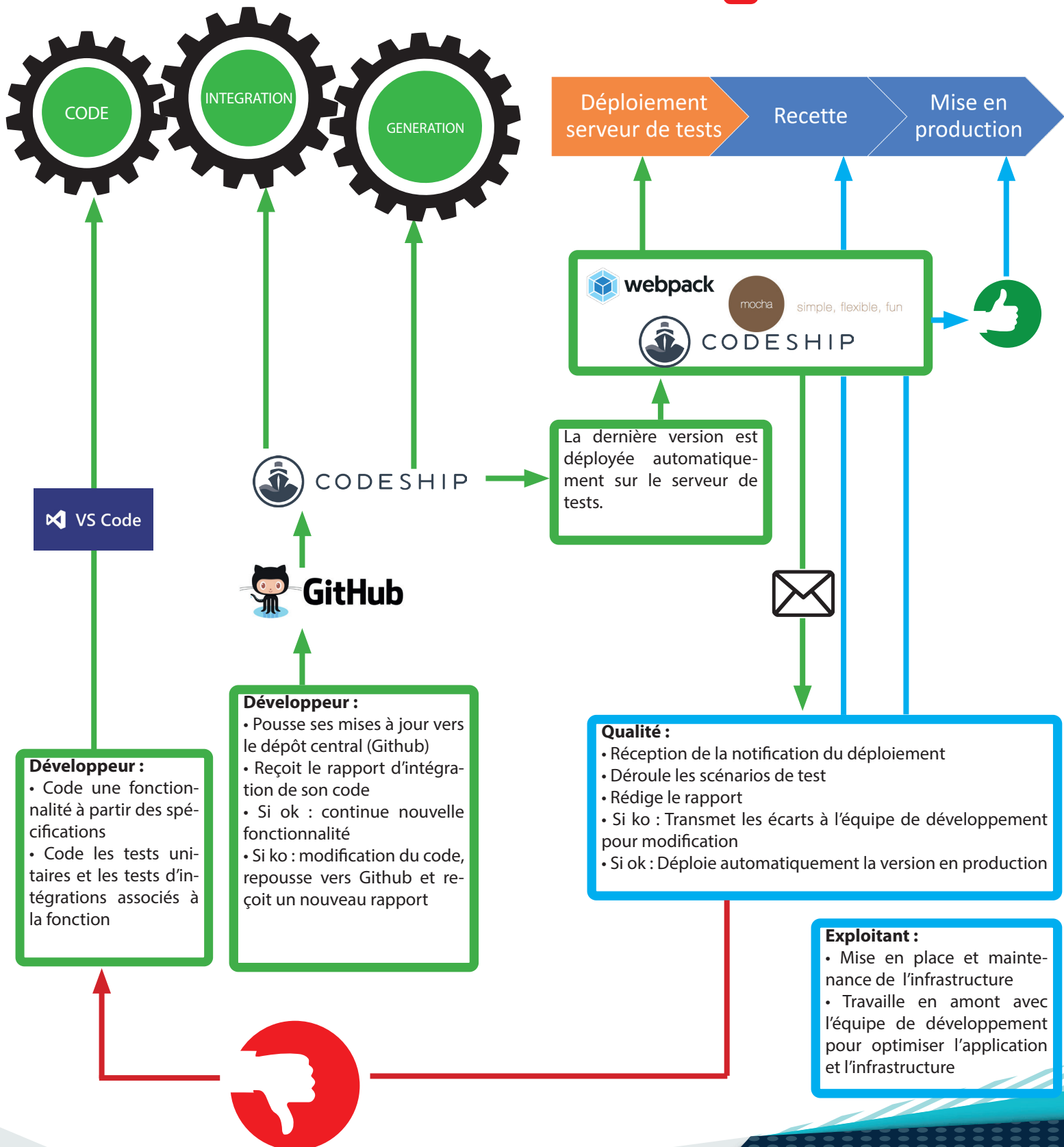
L'intégration continue permet d'automatiser les phases de test lorsque le développeur modifie son code. Les avantages de l'intégration continue sont de trouver et de corriger plus rapidement les bogues, d'améliorer la qualité des logiciels et de réduire le temps nécessaire pour valider et publier de nouvelles mises à jour de logiciels.

Livraison continue

Automatisation

Semi
automatique

Étapes non automatique

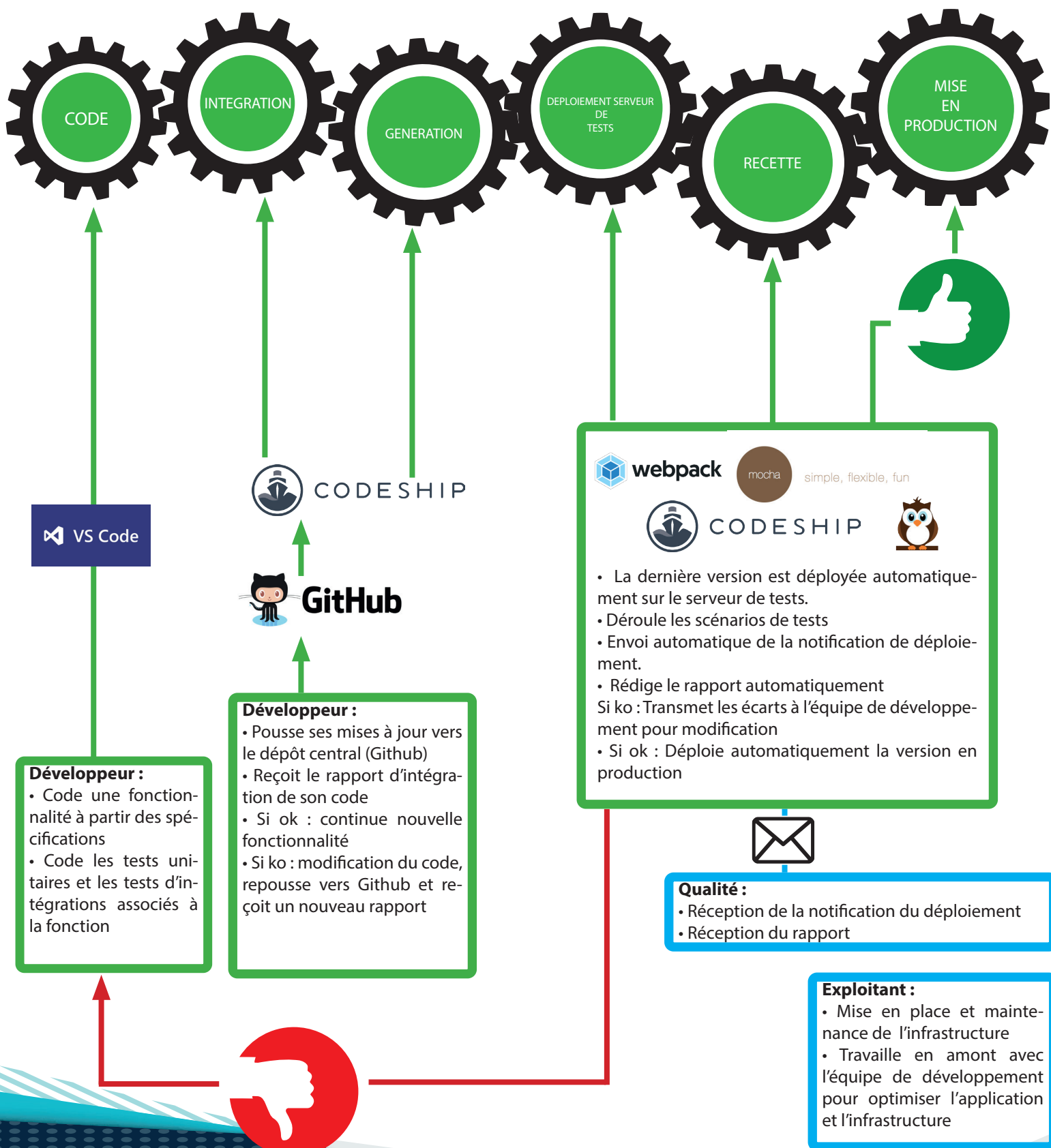


Le saviez-vous ?

La livraison continue permet aux développeurs d'automatiser les tests au-delà des simples tests d'unité, afin de vérifier différents aspects d'une mise à jour d'application avant de la déployer auprès des clients. De cette manière, les développeurs peuvent vérifier de façon plus complète les mises à jour et détecter les éventuels problèmes à corriger avant le déploiement. Une fois tous les tests validés, les développeurs valident la mise en production.

Déploiement continu

Automatisation



Le saviez-vous ?

Le déploiement continu veut dire que tout changement traverse tout le pipeline et soit automatiquement passé en production, entraînant quotidiennement de nombreux déploiements. Pour faire du déploiement continu, vous devez faire de la livraison continue.

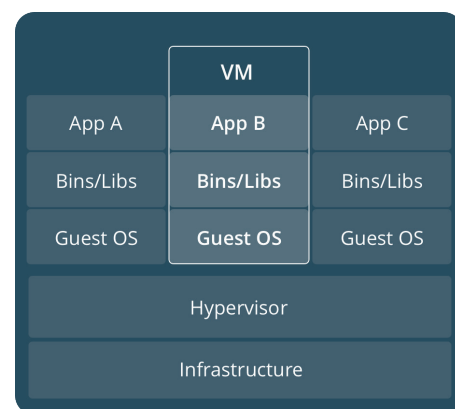
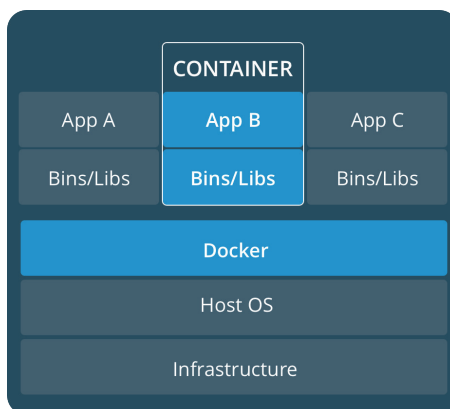
Docker

Docker c'est quoi ?

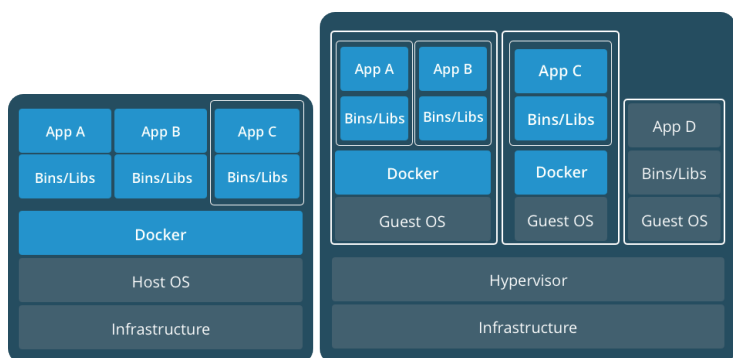
Docker est un logiciel qui permet de stocker une application dans un container virtuel laquelle pourra s'exécuter sur n'importe quelle machine d'un même environnement. Contrairement aux machines virtuelles traditionnelles il n'est pas nécessaire d'avoir un système d'exploitation dédié par application. Les avantages directs sont leur légèreté et leur rapidité d'exécution ainsi que leur facilité d'administration.

Docker permet donc de limiter les lourdes mises à jour applicatives grâce à la possibilité de containeriser l'application : chaque couche applicative peut posséder son propre container.

Dans le cadre d'une mise à jour, seul le container à mettre à jour sera modifié. Dans le cadre du déploiement continu, Docker permet le rapprochement des équipes de DEV et OPS, dans la mesure où les deux travaillent conjointement en amont du déploiement de l'application. Elle permet donc d'accélérer le cycle de déploiement des applications.



A la différence des machines virtuelles qui utilisent plusieurs systèmes d'exploitations, Docker utilise le même noyau système pour tous les containers.
Source image docker.com



Les conteneurs et les machines virtuelles utilisés ensemble offrent une grande flexibilité dans le déploiement et la gestion des applications.

Source image docker.com

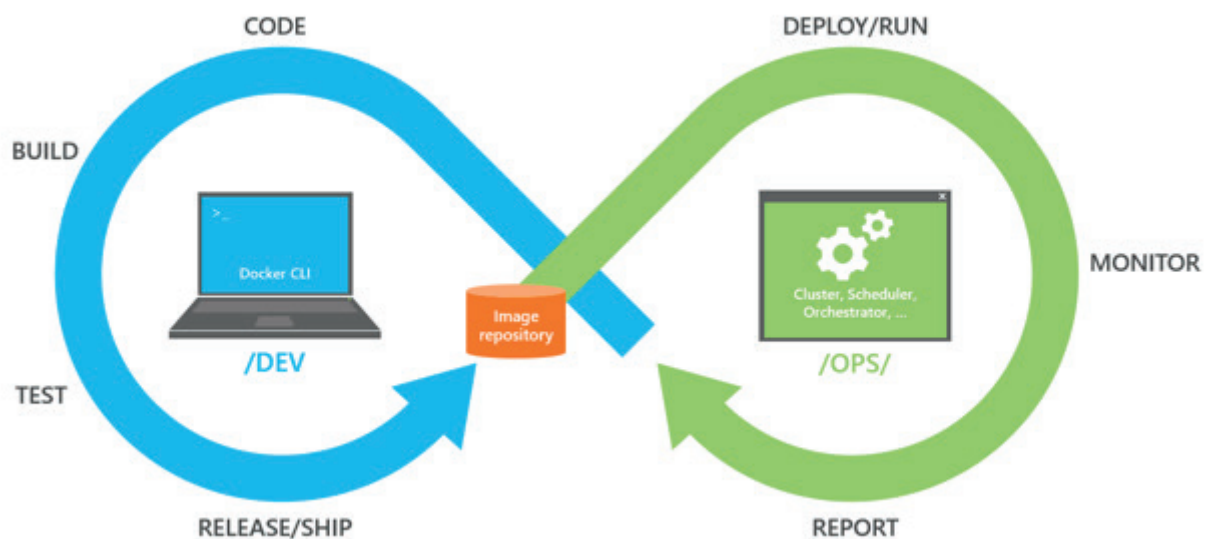
Un peu d'histoire

La technologie des containers existe depuis longtemps. Par exemple, il y a plus de 10 ans, Sun a introduit dans Solaris cette technologie de containers : les Solaris Zones. Puis, cette technologie a été introduite dans le monde Linux avec l'outil LXC qui est à l'origine de Docker.

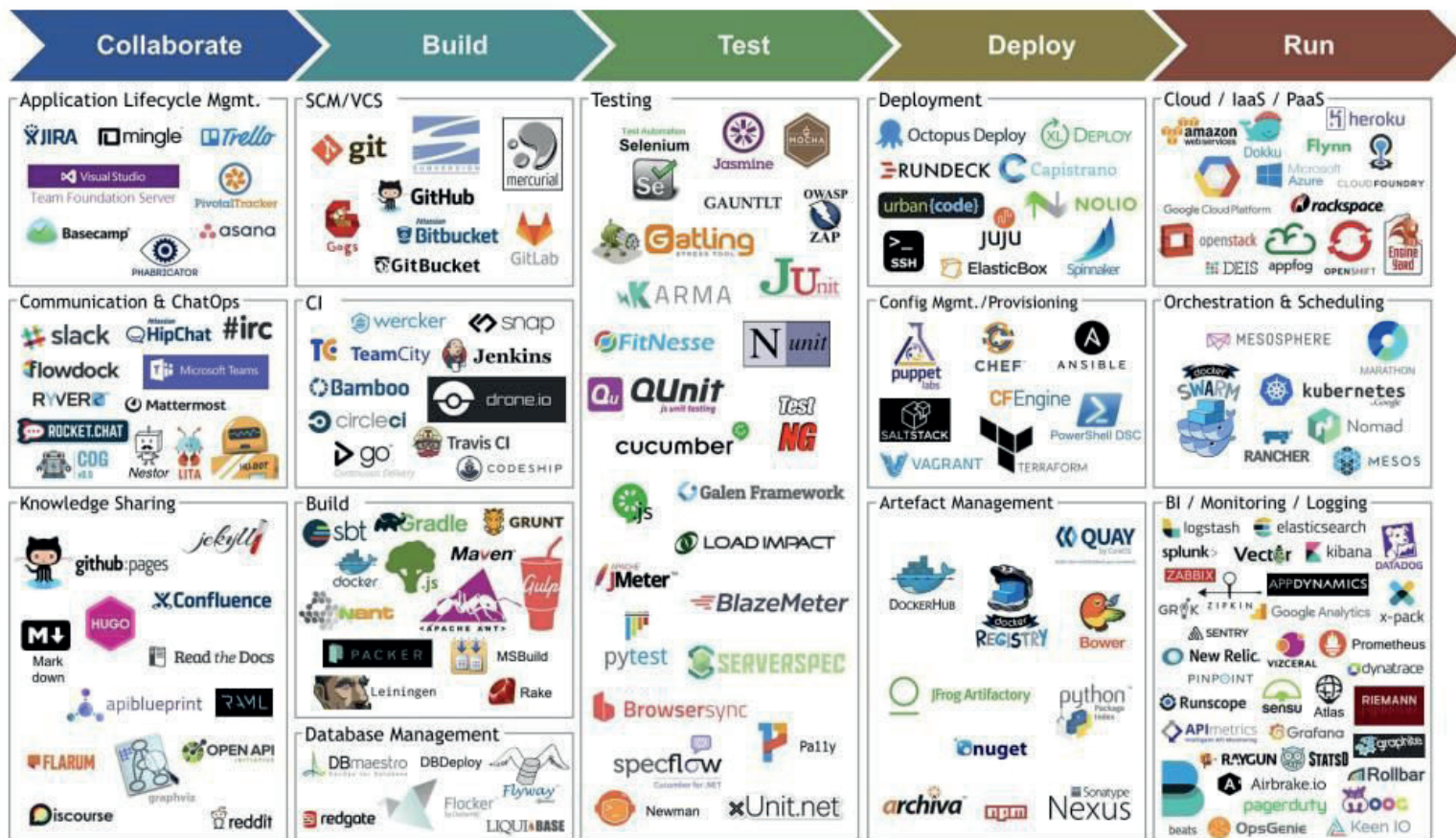
DevOps

DevOps quand le développement rencontre l'exploitation

Le terme DevOps est issu de la contraction de développement (« Dev ») et exploitation (« Ops »). Il s'agit d'une méthode, une culture faite de bonnes pratiques qui permet l'alignement de l'ensemble des équipes du système d'information sur un objectif commun, celui de répondre aux besoins des clients. L'un des objectifs du mouvement DevOps étant de pouvoir s'adapter rapidement aux demandes et aux changements du business.



Les outils qui contribuent au mouvement DevOps sont nombreux, en voici quelques exemples ci-dessous :



Remerciements

Merci à Olivier BUFFAT (responsable de formation MSI du CESI de Labège) et à l'équipe pédagogique du CESI pour leurs soutiens et de nous avoir donné les outils nécessaires pour réaliser cette conférence.

Source des images :

Wikipedia.fr
Docker.fr
Pinterest.fr
Freepik.fr