



## AGILE BASICS

### Continuous improvement

En tant qu'équipe (Ops et support compris), nous travaillons ensemble à améliorer notre fonctionnement en continu.

Par exemple, des rétrospectives sont faites régulièrement. Les actions font l'objet d'un suivi régulier, mesurable et officiel

*#CI*



## AGILE BASICS

### Backlog management

Le Product Owner met régulièrement à jour un backlog ordonné et qualifié, qui est partagé avec l'équipe et l'ensemble des parties prenantes. Avant le développement, nous, l'équipe en entier, découpons en User Stories, et travaillons ensemble sur les critères d'acceptance. Il existe une DoR et une DoD

*#grooming #backlog*



## AGILE BASICS

### Synchronisation

A chaque niveau de l'organisation, les points de synchronisation sont structurés et connus de tous : l'équipe fait des daily meetings, des backlog groomings, des sessions inter-équipes, des démonstrations, et s'appuie sur un visual management efficace et clair

*#daily #demo #ritual*



## METROLOGY

### Business indicators

Afin de piloter son activité et améliorer son efficacité opérationnelle, le business dispose d'indicateurs métier dans des tableaux de bord, mis à jour de façon automatique

*#value*



## METROLOGY

### Strategy & tools

Nous (Business et IT) disposons d'une stratégie commune permettant d'améliorer notre capacité globale à mesurer grâce à des outils permettant d'exposer l'évolution d'indicateurs dans le temps et d'effectuer des corrélations

*#datalake #ELK*



## METROLOGY

### Usage tracking

Afin de détecter des anomalies, monitorer l'activité et l'usage du produit, nous (tous les acteurs IT) avons accès à des tableaux de bord partagés mis à jour en temps réel

*#monitoring*



## CD PIPELINE

### Build & Package

La construction et le packaging des artefacts sont réalisés par un automate. Ils sont déposés automatiquement sur un serveur de stockage sécurisé. La traçabilité est assurée

*#continuousIntegration*

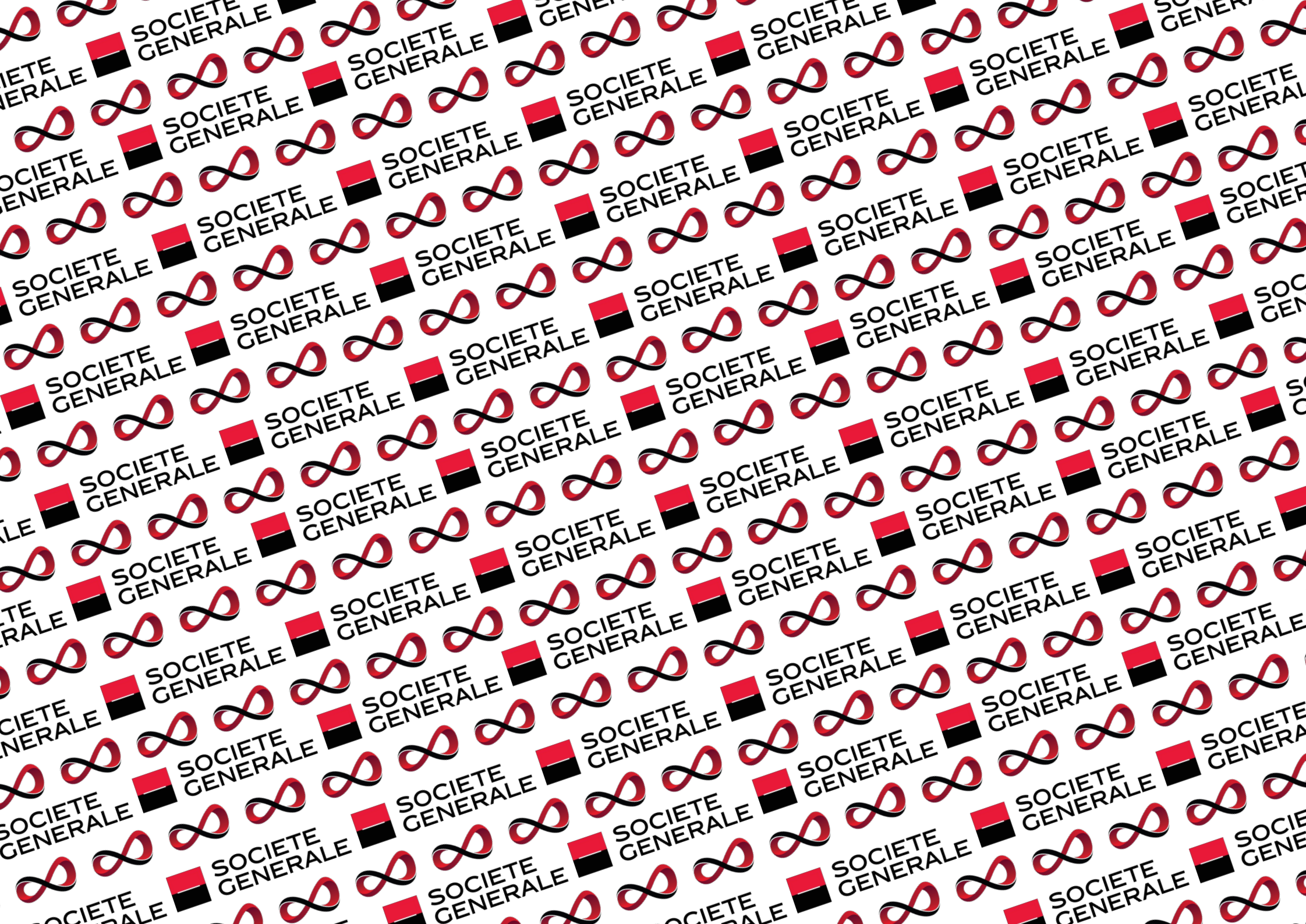


## CD PIPELINE

### Provisionnement & Deploy

Afin de réduire les erreurs et le temps de déploiement, la mise à disposition d'environnements et le déploiement des livrables sont réalisés par un automate. Dev et Ops partagent les mêmes outils, qui sont utilisés depuis Dev jusque Prod

*#infraAsCode #IAAS*





## CD PIPELINE

### Continuous tests

Afin de fournir un feedback rapide sur les développements et de réduire la durée de test malgré l'augmentation de la base de code de l'application, les tests sont déclenchés automatiquement dans l'usine logicielle

*#fastFeedback #testAuto*



## TESTING

### Strategy

Nous maintenons à jour notre stratégie de test, qui permet à tous de visualiser les enjeux et les moyens mis en œuvre pour assurer la qualité de l'acceptance du produit

*#testPyramid*



## TESTING

### Frameworks & Tools

Nous disposons de frameworks et outils de tests permettant de répondre aux différents types de tests, et ceux-ci sont adaptés au contexte technique et fonctionnel

*#unitTest #intTest #perf*



## TESTING

### Practices

Nous connaissons les pratiques du TDD et BDD, et les typologies de tests (unitaire, intégration, E2E, performance). Nous choisissons et maintenons la meilleure technique en fonction du contexte

*#TDD #BDD #testFirst*



## CODE QUALITY

### Practices

Les développeurs maîtrisent les principes théoriques de la qualité de code : CleanCode, Legacy refactoring, SOLID. Les développeurs appliquent la Boy Scout rule sur la base de code

*#cleanCode #refactoring*



## CODE QUALITY

### Community

Les développeurs font régulièrement des revues de code, lors desquelles ils peuvent échanger, confronter leurs points de vue, et faire évoluer les règles communes, qui sont formalisées et partagées de façon transparente et accessible

*#codeReview #pair*



## CODE QUALITY

### Tools

Des outils d'analyse de code permettent de mesurer la qualité de façon tangible, de pointer les violations de règles, et de constater l'évolution dans le temps des métriques

*#sonar*



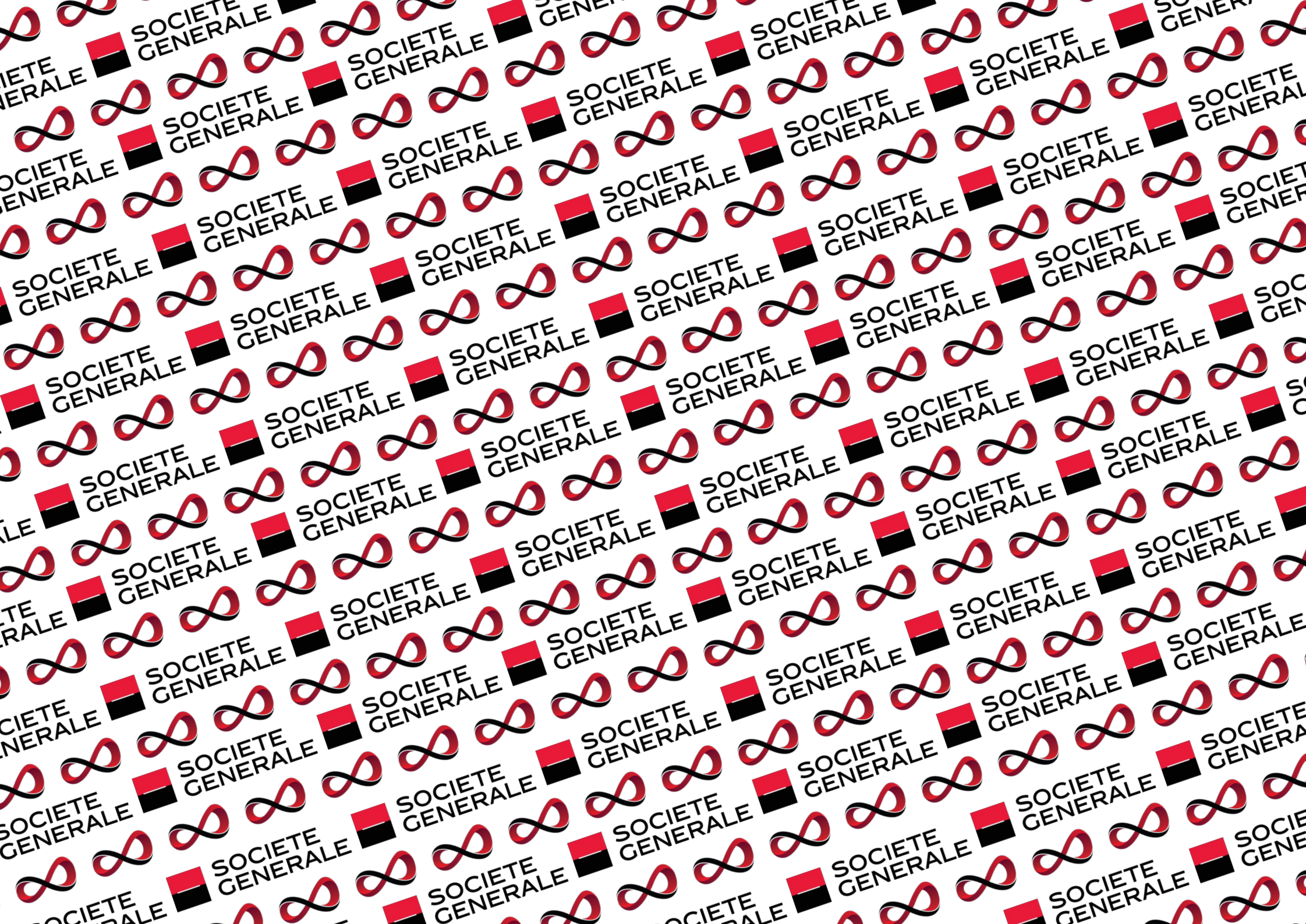
## COLLABORATION

### Transparency

Afin d'avoir une efficacité globale, nous nous attachons à exposer à nos partenaires nos objectifs, nos contraintes et nos priorités. La transparence réciproque permet d'assurer une meilleure collaboration

*#systemThinking*







## COLLABORATION

### Collective ownership

En tant qu'équipe, nous portons une attention particulière à notre résilience. Nous sommes co-responsables et assurons la pérennité de nos pratiques en évitant de polariser rôles et connaissances sur une seule personne

*#oneTeam #noHero*



## COLLABORATION

### Spec by example

Specs, tests et guides utilisateurs sont un seul et même document, produit par l'ensemble des parties prenantes qui explorent les besoins métiers et prennent des exemples selon une démarche itérative. Cette documentation vivante supporte les tests automatiques et fait gagner du temps

*#BDD #executableDoc*



## PARTNERSHIP

### PO onboarded

Le Product Owner fait partie de l'équipe, il écoute les membres, il accepte de faire des expérimentations. Il contribue à l'amélioration continue en participant notamment aux rétrospectives

*#productOwnership*



## PARTNERSHIP

### Business value model

Afin de maximiser la valeur délivrée, nous (business et IT) utilisons un format de conversation efficace et un vocabulaire communs lorsque nous priorisons les changements. La réflexion se base sur les objectifs métier, et intègre: contexte, coûts, bénéfices, risques, qualité...

*#value #BVM*



## PARTNERSHIP

### Slicing

Une attention particulière est portée sur les découpages, ce qui nous permet de livrer régulièrement des incréments apportant de la valeur aux clients. Tous les membres de l'équipe (PO compris) prennent part régulièrement à des sessions de découpage

*#storyMapping*



## LEADERSHIP

### Decentralized decisions

Afin d'éviter les goulots d'étranglement et le contrôle centralisé, les ordres imposés par la hiérarchie sont limités. Les responsables expriment clairement les domaines où la décision est hiérarchique et ceux où elle laisse l'équipe autonome

*#delegationPoker*



## LEADERSHIP

### Space for sharing

Les managers cherchent à faciliter la communication et les échanges, la montée en compétences des équipes (interne+externe). Du temps est consacré à l'échange et au partage : BBL, parcours pour les nouveaux arrivants, binomage avec mentor, mise en avant des membres les plus pédagogues, etc...

*#BBL #hackathon*



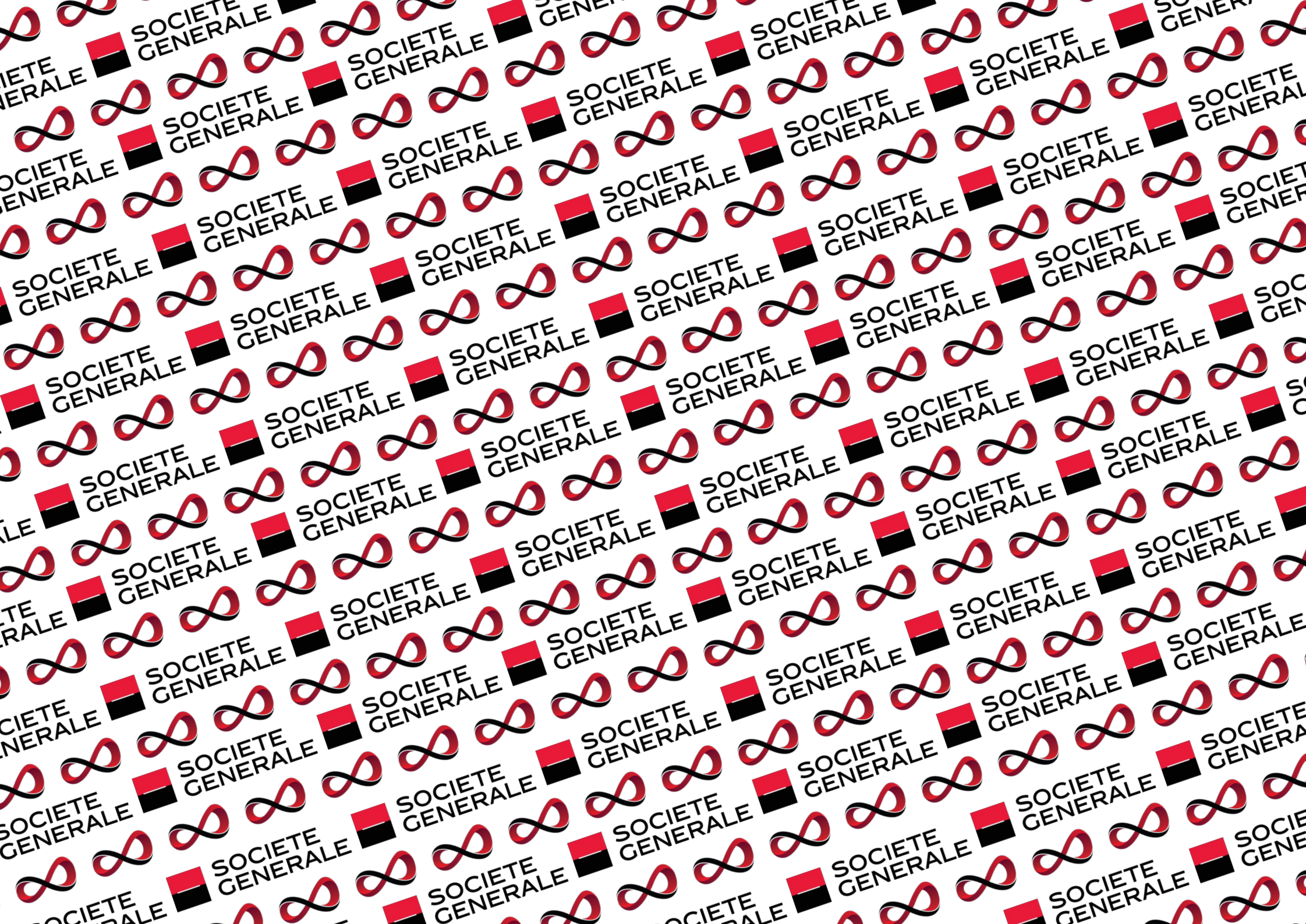
## LEADERSHIP

### Start with why

Plutôt que de prendre des décisions sur le « quoi » ou le « comment », les managers exposent l'intention et la vision, de telle sorte que l'équipe puisse proposer les solutions en toute confiance

*#hostLeadership*







## PEOPLE

### Develop People

La confiance fait partie intégrante du mode de fonctionnement. Le feedback constructif régulier se fait naturellement, sans crainte de jugement. Les compétences de chacun sont connues et des parcours de formation individuels existent

*#trust #training*



## PEOPLE

### Respect

L'expérimentation et le droit à l'erreur sont des principes forts de l'équipe. Les remontées de problèmes sont valorisées car source d'amélioration continue. On s'attache à un haut niveau d'exigence afin de fournir aux partenaires un produit de la meilleure qualité possible

*#rightForFailure*



## PEOPLE

### Role modeling

Des règles de vie / valeurs existent, sont construites et partagées par l'équipe (y compris manager). Plutôt que de donner des leçons, chacun s'efforce de montrer l'exemple

*#leadByExample*



## OPERATION

### License to prod

Tout membre de l'équipe a réussi à obtenir la licence de production avant d'accéder à la production via la solution informatique IAM utilisée par mon application

*#production #licensetoprod*



## OPERATION

### Event, incident, problem

Un event permet aux équipes d'analyser et résoudre de manière proactive les incidents avant qu'ils n'affectent les utilisateurs. Un incident peut perturber ou réduire la qualité d'un service et impacte les activités des utilisateurs. Un problème est la cause fondamentale d'un ou plusieurs incidents

*#event #incident #problem*



## OPERATION

### Continuity of service

Les besoins en assistance informatique métier sont gérés selon une configuration définie, anticipée et communiquée, y compris les événements récurrents (expiry, weekend, infrastructure, jours fériés) et non récurrents (élections générales)

*#infraday  
#majorchangeweek*



## BEST PRACTICES

### Practices & processes

L'activité de production est suivie et trackée dans les outils adéquats suivant les pratiques et processus IT et les différents tickets sont visibles et priorisés dans le backlog de l'équipe

*#process #itsm #itil  
#impulse*



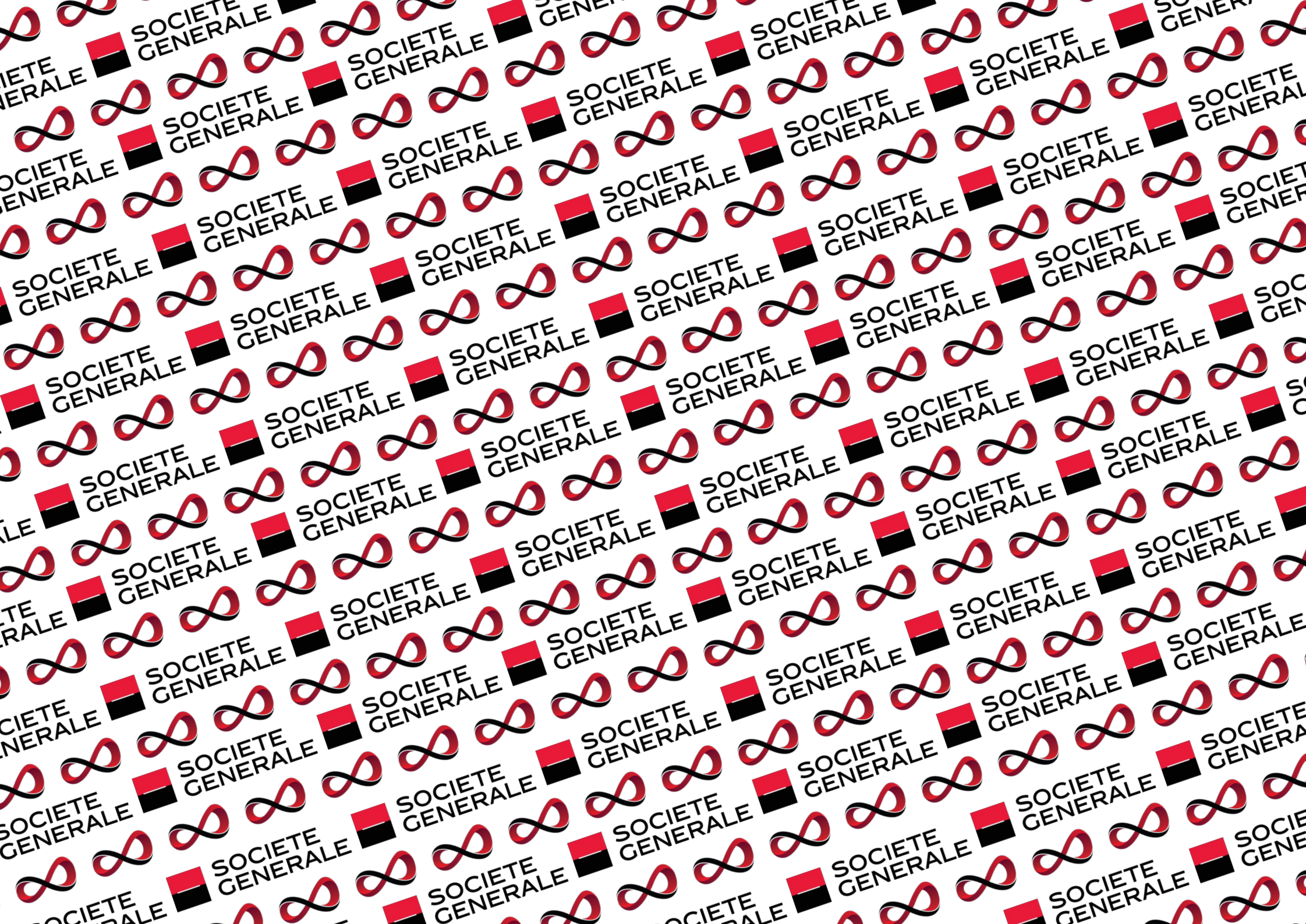
## BEST PRACTICES

### Support & process

En fonction de mes besoins métiers, je m'assure qu'une équipe de support est en mesure de répondre aux demandes et de gérer les incidents sur leurs heures ouvrables (en fonction du SLA)

*#production #followthesun  
#sla*









## BEST PRACTICES

### **Knowledge management**

Je m'assure que les connaissances critiques de l'exécution des activités de production sont formalisées dans des documents partagés entre les équipes de production du monde entier

*#knowledge #documents*



## CONTRACT

### **SLA**

Je connais les processus métiers auxquels mon périmètre applicatif contribue et les accords de niveau de service associés. Je participe au processus de révision de ce contrat SLA (Service Level Agreement)

*#sla #niveauxdeservice*

