

Tutoriel :

IA pour les tests

Apprendre à prompter

- Bruno Legeard
- Fabrice Bouquet
- Abbas Ahmad
- Anne Kramer
- Alexandre Vernotte

Plan du tutoriel

Requêter une IA conversationnelle pour les tests

- 1 - Introduction : IA générative pour les tests logiciels
 - Atelier 0 - Prise en main du portail LLM
- 2 - Techniques de Prompting
- 3 - Ateliers
 - Atelier 1 - Génération de tests
 - Atelier 2 - Priorisation des tests
 - Atelier 3 - Automatisation des tests
- 4 - Bonnes pratiques du prompting pour les tests logiciels
- 5 - Bilan et discussion sur les savoir-faire mis en oeuvre



Structure des ateliers

Introduction

Mise en pratique - Prompt Engineering

Atelier 0 - Prise
en main du
portail LLM

Atelier 1 - Génération
de tests

Atelier 2 -
Priorisation des tests

Atelier 3 -
Technique RAG

Ce tutoriel utilise des extraits de la formation "**Accélérer les processus de test grâce à l'IA générative**" par la pratique, conçue sur 2 jours pour la profession Testeur.



Matériel fourni & outils utilisés

Matériel fourni

- Support du tutoriel
- Matériel des ateliers (données et exercices)

Outils utilisés

- LLMs → via notre portail LLM
 - OpenAI GPT-x, Mistral, Meta Llama, Perplexity, Anthropic Claude 3
- RAG → via notre portail LLM

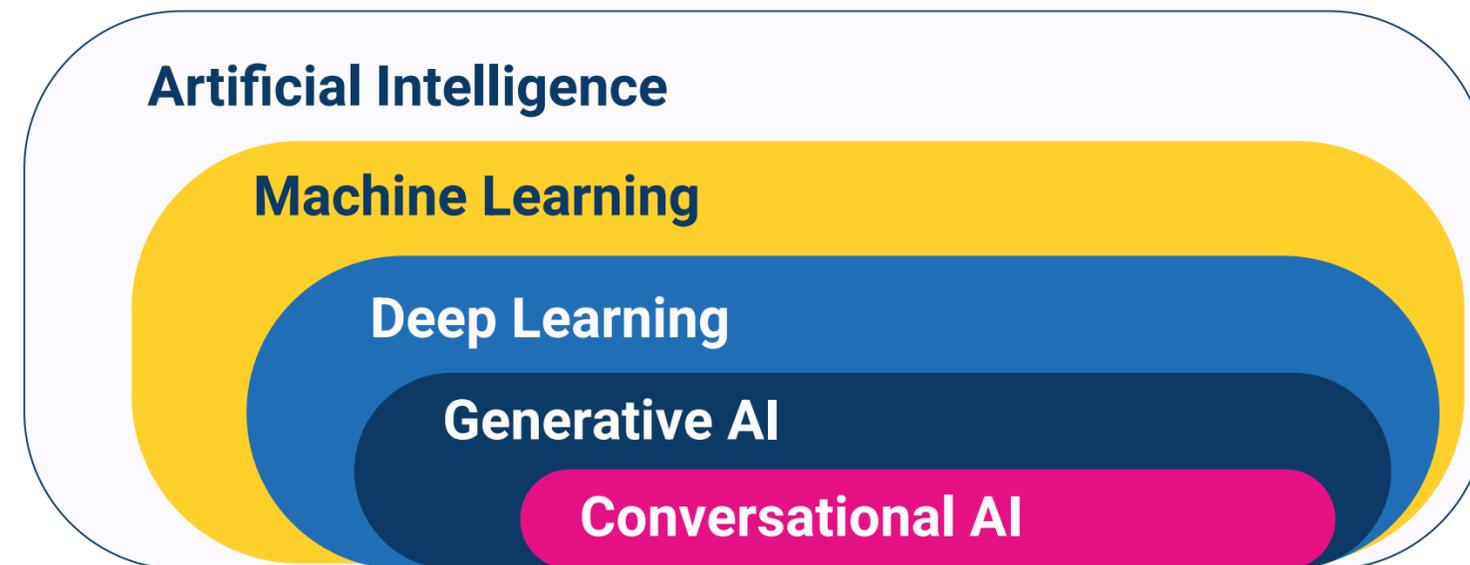
LLM = Large Language Model



1- IA générative - Les bases

Machine learning, deep learning et IA générative

Différents types de technologies IA



L'IA **générative** fait partie des techniques de Deep Learning, avec la particularité d'utiliser des **modèles pré-entraînés** sur de grands corpus de données.

LLM = Large **L**anguage **M**odel



Tester avec l'IA générative

Un impact sur de nombreuses activités de test

🧠 **Interpréter** des specs, de exigences, des images écrans, des cas de test, des rapports d'anomalies, etc...

⚙️ **Générer** des conditions de test, des cas de tests, du code de test, etc...

📄 **Synthétiser** des rapports d'avancement, d'anomalies, de couverture, etc...

✅ **Evaluer / corriger** la couverture de test, le code de test, etc...

1. Planification des tests

2. Suivi et contrôle des tests

3. Analyse de test

4. Conception des tests

5. Implémentation des tests

6. Exécution des tests

7. Clôture des tests

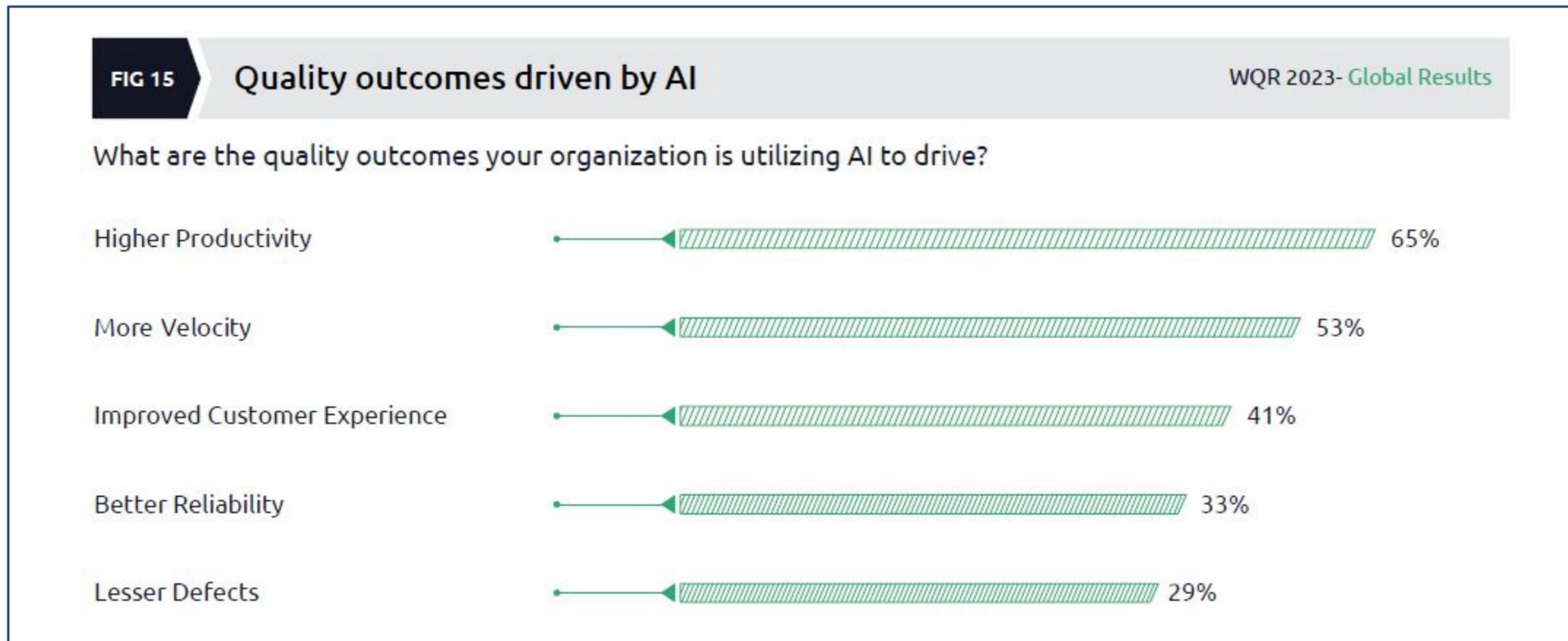


IA générative & Tests

- Analyser et améliorer les exigences
 - Identifier les ambiguïtés, les incohérences et les informations manquantes
 - Générer des questions pertinentes pour guider les discussions avec les parties prenantes
- Aider à la conception des tests
 - Proposer des idées de cas de test pertinents
 - Générer des scénarios de test, des conditions limites et des combinaisons de données de test
- Aider à l'automatisation des tests
 - Générer du code de test automatisé à partir de descriptions en langage naturel
 - Proposer des améliorations au code existant
 - Identifier les modèles de conception de test appropriés
 - Générer des scripts de test pour différents frameworks d'automatisation
- Faciliter la maintenance des tests
 - Identifier les doublons
 - Suggérer des optimisations
 - Mettre à jour les tests obsolètes suite aux changements de spécifications
 - Générer de la documentation de test actualisée
- Générer des données de test
 - Créer des jeux de données de grande taille
 - Générer des cas de test aux limites
 - Produire des données synthétiques pour les tests de performance et de sécurité
- Analyser les résultats d'exécution
 - Identifier les modèles d'échec
 - Suggérer des causes potentielles
 - Générer des rapports de synthèse
 - Classer et hiérarchiser les défauts en fonction de leur sévérité et de leur impact
- Créer de la documentation
 - Générer des plans de test
 - Produire des rapports de test
 - Créer des guides utilisateur
 - Tenir à jour la documentation en fonction des changements apportés au système et aux processus de test

Des attentes fortes sur l'IA

Productivité et vélocité en tête des attentes des organisations



Source : Word Quality Report - 2023-24



Utiliser l'IA générative pour les tests

Comment utiliser un LLM ?

Les échanges sont toujours réalisés sous forme de conversation

Requête (Prompt) ➔ Réponse du LLM

- **Via Chatbot** : plus convivial pour converser / interroger directement le LLM
- **Via API** : plus souple et configurable. Permet de développer des applications basées LLM

Bien utiliser un LLM implique de maîtriser le **PROMPT ENGINEERING**



Prompt Engineering

Signification du terme “Prompt Engineering”

Prompt Engineering fait référence à la conception de requêtes de qualité qui guident les modèles LLM pour produire des sorties précises.

💡 Cela implique de sélectionner le bon **type de requêtes**, d'**optimiser leur longueur** et **leur structure**, et de déterminer leur **ordre** et leur **pertinence** par rapport à la tâche.

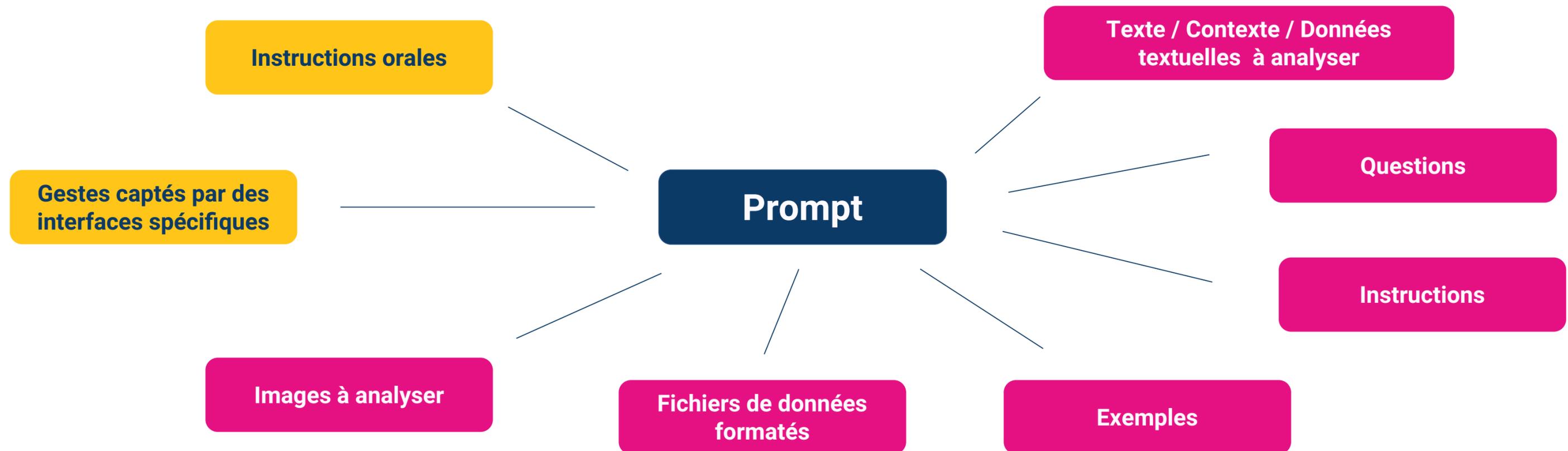
👉 Vos résultats vont dépendre du modèle LLM utilisé et de vos prompts

Prompt Engineering = Travailler la structure, le style et le contenu des requêtes



Que trouve-t-on dans un “prompt” ?

Un prompt peut être multimodal



Prompt Engineering pour une tâche

Le Prompt Engineering n'est pas universel, mais doit correspondre aux types d'activités à réaliser.

Marketing - Prompts pour la production de messages & slogans

- Accent sur la **créativité**, la voix de la marque et la **persuasion**
- Requêtes conçues pour générer un contenu qui
 - Résonne avec un **public cible** spécifique
 - S'aligne sur **l'identité de la marque**
 - **Incite à l'action**
 - Intègre le **référencement et les mots-clés**

Tests logiciels - Prompts pour la génération de cas de test

- Accent sur la **précision**, **l'exhaustivité** et la garantie que **tous** les scénarios possibles sont testés
- Requêtes très spécifiques et techniques
 - Incluant des **descriptions détaillées** des fonctionnalités à tester
- Langage **plus structuré** et moins axé sur la créativité que dans le domaine du marketing



Structure de base du prompt



Le prompt type contient 6 parties :

Rôle

Quel rôle (par exemple “concepteur de tests”) est assigné au LLM

Contexte

- Décrire succinctement le contexte de la tâche

Instructions

- Ce qu’il faut réaliser - éventuellement en plusieurs étapes

Format

- Le format de sortie attendu

Contraintes (éventuelles)

- Pour préciser des limitations ou exigences de réalisation

Données (éventuelles)

- Exemples de réponse
- Données de la tâche à réaliser



Exemple de prompt structuré

Utiliser la structure Rôle-Contexte-Instructions-Contraintes-Format-Données

Tu es mon assistant pour l'écriture de scénarios de test en format Gherkin (Given/When/Then) à partir d'une exigence donnée. Le système sous test est l'application web Logogénie (voir <https://www.logogenie.fr/>).

En analysant cette exigence, réalise les tâches suivantes :

1. Détermine de manière exhaustive les classes d'équivalence à tester, en donnant quelques exemples de valeurs typiques et aux limites pour les classes valides et invalides. Explique ton raisonnement.
2. Crée les scénarios dans un fichier <nom_feature>.feature, en regroupant les scénarios par fonctionnalité. Assure-toi que toutes les classes d'équivalence sont couvertes avec au moins 2 tests chacune.

#

Exigence :

La création du logo nécessite que les champs suivants soient saisis : nom de l'entreprise (obligatoire, 40 caractères max), phrase de slogan (50 caractères max, optionnel), et mots-clés (au moins un).

 Analyser ce prompt et trouver les éléments de la structure



Analyse du prompt



Tu es mon **assistant pour l'écriture de scénarios de test en format Gherkin (Given/When/Then)** à partir d'une exigence donnée. Le **système sous test est l'application web Logogénie (voir <https://www.logogenie.fr/>)**.

En analysant cette exigence, réalise les tâches suivantes :

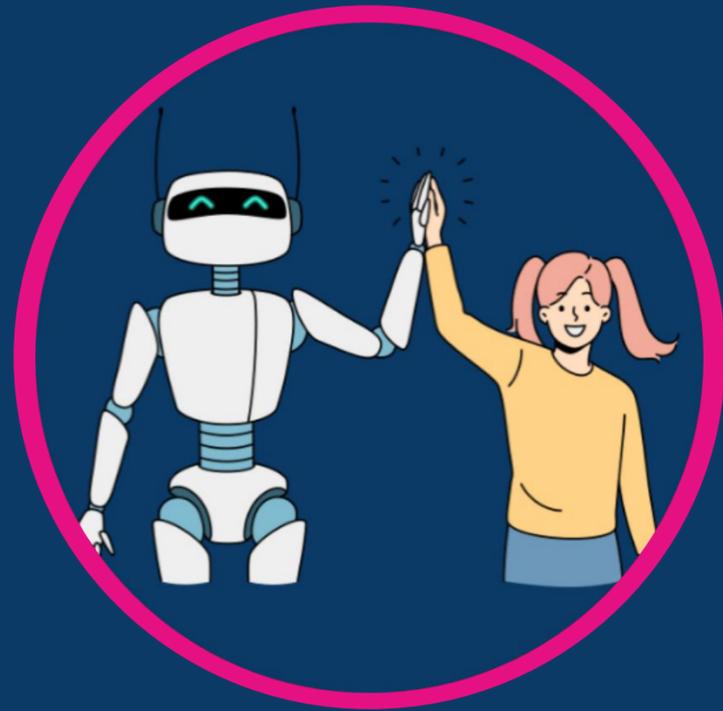
1. Détermine de manière exhaustive les classes d'équivalence à tester, en donnant quelques exemples de valeurs typiques et aux limites pour les classes valides et invalides. Explique ton raisonnement.
2. Crée les scénarios dans un fichier **<nom_feature>.feature**, en regroupant les scénarios par fonctionnalité. Assure-toi que toutes les classes d'équivalence sont couvertes avec **au moins 2 cas de tests chacune**.

#

Exigence :

- La création du logo nécessite que les champs suivants soient saisis : nom de l'entreprise (obligatoire, 40 caractères max), phrase de slogan (50 caractères max, optionnel), et mots-clés (au moins un).





Tutoriel :

IA pour les tests

Apprendre à prompter

16

Atelier 0
Découverte du portail LLM

Mise en contexte de cet atelier



Contexte

- Vous êtes Ingénieur QA au sein d'un éditeur de logiciels dans le domaine de la communication et du marketing.
- Une application Web de création de logo a été rachetée par votre entreprise. Votre entreprise souhaite réaliser une migration technique vers .NET, qui est la plateforme d'entreprise.
- Il a été constaté qu'aucun test n'avait été fourni lors de la livraison du code : il vous est demandé de développer une suite de tests manuels qui vise à s'assurer de l'absence de régressions lors de la migration technique.

Pourquoi utiliser de l'IA pour vous aider

- Vous avez très peu de temps pour cette tâche et vous souhaitez utiliser un assistant IA pour la création des cas de test et leur mise en forme.





Réalisation de l'atelier

Ce que vous allez faire

- Utiliser un prompt structuré
- Comparer le comportement de deux LLM

et aussi...

- Découvrir le portail LLM Smartesting utilisé pour le tutoriel

Les consignes sont données sur la page : “JFTL 2024 - Tutoriel IA - Atelier 0” -
www.cftl.fr/tuto-ia



Plan du tutoriel

Requêter une IA conversationnelle pour les tests

- 1 - Introduction : IA générative pour les tests logiciels
 - Atelier 0 - Prise en main du portail LLM
- ● 2 - Techniques de Prompting
- 3 - Ateliers
 - Atelier 1 - Génération de tests
 - Atelier 2 - Priorisation des tests
 - Atelier 3 - Automatisation des tests
- 4 - Bonnes pratiques du prompting pour les tests logiciels
- 5 - Bilan et discussion sur les savoir-faire mis en oeuvre



2- Techniques du Prompting

Principaux types de requêtes

Requêtage structuré

Ajouter du contexte, des instructions, des exemples (one-shot / few-shot)

Requêtage par étapes

Découper votre demande en étapes intermédiaires - Chain-of-Thoughts

Requêtage non-structuré en cascade

Dans le style d'une conversation humaine avec une suite d'échange de questions / réponses et rebond sur la question suivante

Méta-requêtage et évaluation

Utiliser un LLM pour créer ou améliorer vos requêtes
Demander d'évaluer votre prompt



Requêtage structuré

Utiliser la structure Contexte-Rôle-Instruction-Contraintes-Format-Données

Tu es mon assistant pour l'écriture de scénarios de test en format Gherkin (Given/When/Then) à partir d'une exigence donnée. Le système sous test est l'application web Logogénie (voir <https://www.logogenie.fr/>).

En analysant cette exigence, réalise les tâches suivantes :

1. Détermine de manière exhaustive les classes d'équivalence à tester, en donnant quelques exemples de valeurs typiques et aux limites pour les classes valides et invalides. Explique ton raisonnement.
2. Crée les scénarios dans un fichier <nom_feature>.feature, en regroupant les scénarios par fonctionnalité. Assure-toi que toutes les classes d'équivalence sont couvertes avec au moins 2 tests par classe.

#

Exigence :

La création du logo nécessite que les champs suivants soient renseignés : nom de l'entreprise, adresse, numéro de téléphone, slogan (50 caractères max, optionnel), et mots-clés (au moins un).

Etre cohérent dans les termes utilisés.



Requêtage structuré

Utiliser la structure Rôle-Contexte-Instructions-Contraintes-Format-Données

Tu es mon assistant pour l'écriture de scénarios de test en format Gherkin (Given/When/Then) à partir d'une exigence donnée. Le système sous test est l'application web Logogénie (voir <https://www.logogenie.fr/>).

En analysant cette exigence, réalise les tâches suivantes :

1. Détermine de manière exhaustive les classes d'équivalence à tester, en donnant quelques exemples de valeurs typiques et aux limites pour les classes valides et invalides. Explique ton raisonnement.
2. Crée les scénarios dans un fichier <nom_feature>.feature, en regroupant les scénarios par fonctionnalité. Assure-toi que toutes les classes d'équivalence sont couvertes avec au moins 2 tests chacune.

#

Exigence :

La création du logo nécessite que les champs suivants soient saisis : nom de l'entreprise (obligatoire, 40 caractères max), phrase de slogan (50 caractères max, optionnel), et mots-clés (au moins un).

Répéter ce qui est important pour augmenter le focus.

Requêtage structuré

Utiliser la structure Rôle-Contexte-Instructions-Contraintes-Format-Données

Tu es mon assistant pour l'écriture de scénarios de test en format Gherkin (Given/When/Then) à partir d'une exigence donnée. Le système sous test est l'application web Logogénie (voir <https://www.logogenie.fr/>).

En analysant cette exigence, réalise les tâches suivantes :

1. Détermine de manière exhaustive les classes d'équivalence à tester, en donnant quelques exemples de valeurs typiques et aux limites pour les classes valides et invalides. Explique ton raisonnement.
2. Crée les scénarios dans un fichier <nom_feature>.feature, en regroupant les scénarios par fonctionnalité. Assure-toi que toutes les classes d'équivalence sont couvertes avec au moins 2 tests chacun.

#

Exigence :

La création du logo nécessite que les champs suivants soient saisis : nom de l'entreprise, slogan (50 caractères max, optionnel), et mots-clés (au moins un).

Intégrer des demandes d'explication et d'auto-vérification.

Requêtage structuré

Utiliser la structure Rôle-Contexte-Instructions-Contraintes-Format-Données

Tu es mon assistant pour l'écriture de scénarios de test en format Gherkin (Given/When/Then) à partir d'une exigence donnée. Le système sous test est l'application web Logo génie (voir <https://www.logogenie.fr/>)

En analysant

1. Déterminer les données de valeurs typiques et aux limites
2. Créer les scénarios de test en fonction de la connaissance de l'application. Assure-toi que toutes les classes d'objets sont couvertes.

Éviter des erreurs de formulation pouvant créer une mauvaise interprétation - Bien relire le prompt !

- "Tu es tuteur dans une orga industrielle"
- vs. "Tu es testeur dans une organisation industrielle."

#

Exigence :

La création du logo nécessite que les champs suivants soient saisis : nom de l'entreprise (obligatoire, 40 caractères max), phrase de slogan (50 caractères max, optionnel), et mots-clés (au moins un).



Prompt avec exemple

Zero-shot prompting

Écrit des scénarios de test pour un processus d'authentification.

One-shot prompting

Écrit des scénarios de test pour un processus d'authentification.

Exemple :

** TC1 : Informations d'identification valides

Action : Saisir un nom d'utilisateur et un mot de passe valides

Résultat attendu : L'utilisateur se voit accorder l'accès au système

Few-shot prompting

Ajouter plusieurs exemples pour indiquer plusieurs schémas de réponse attendus.



mistral-large-latest

Voici quelques scénarios de test supplémentaires pour un processus d'authentification :

** TC2 : Informations d'identification invalides

Action : Saisir un nom d'utilisateur ou un mot de passe incorrect

Résultat attendu : Le système affiche un message d'erreur et refuse l'accès

** TC3 : Nom d'utilisateur vide

Action : Laisser le champ du nom d'utilisateur vide et saisir un mot de passe

Résultat attendu : Le système affiche un message d'erreur et refuse l'accès

** TC4 : Mot de passe vide

Action : Saisir un nom d'utilisateur et laisser le champ du mot de passe vide

Résultat attendu : Le système affiche un message d'erreur et refuse l'accès

** TC5 : Compte verrouillé

Action : Saisir le nom d'utilisateur et le mot de passe d'un compte verrouillé

Résultat attendu : Le système affiche un message d'erreur et refuse l'accès

** TC6 : Réinitialisation du mot de passe

Action : Cliquer sur le lien de réinitialisation du mot de passe et suivre les étapes de réinitialisation

Résultat attendu : Le système permet à l'utilisateur de réinitialiser son mot de passe et d'accéder au système avec le nouveau mot de passe



Requêtage par étapes

Découpage de la tâche en “bouchées digérables”

Comme un humain, une IA a du mal à faire **plusieurs étapes à la fois**. Il est donc préférable de **couper la tâche en sous-tâches**.

Exemple

1. **Analyse les exigences**. Pour chaque exigence, écris une description courte. Réponds avec un tableau : | ID | nom | description |
2. Pour chaque exigence, **détermine les critères d'acceptation**. Assure-toi que toutes les classes d'équivalences sont couvertes, y compris...
3. Pour chaque exigence, **écrit les scénarios de test nécessaires pour couvrir toutes les critères d'acceptation**. Réponds dans le format...



Requêtage non-structuré en cascade

Conversation du style "Question/Réponse"

 You
Explique ce qu'est un modèle d'IA générative multimodal.

 **mistral-large-latest**

Un modèle d'IA générative multimodal est un type de modèle d'intelligence artificielle qui est capable de comprendre et de générer des données dans différents formats ou modalités, tels que le texte, l'image, l'audio, et la vidéo. Voici quelques points clés :

- **Compréhension multimodale** : Ces modèles peuvent traiter et comprendre des données dans différentes modalités. Par exemple, ils peuvent comprendre le texte dans une image (comme une pancarte dans une photo de rue) ou la relation entre une légende et une image.
- **Génération multimodale** : Ils peuvent également générer des données dans différentes modalités. Par exemple, ils peuvent générer une description textuelle d'une image (génération de texte) ou créer une image à partir d'une description textuelle (génération d'image).
- **Apprentissage multimodal** : Ces modèles sont entraînés sur de larges ensembles de données qui incluent différentes modalités. Cela leur permet d'apprendre les relations et les corrélations entre différentes modalités.

 You
Explique comment est réalisé l'apprentissage multimodal.

Prochaine question en fonction de la réponse du LLM.



Méta-requêtage

Demander au LLM de créer la structure du Prompt souhaité



You

Ecris un prompt détaillé pour qu'un LLM génère des cas de test fonctionnels à partir d'une User Story et de critères d'acceptation.



You

Propose plusieurs variantes de ce prompt et indique quel prompt donnera probablement les meilleurs résultats.

Prompt de vérification du prompt

Evaluation du prompt

Demander au LLM d'évaluer le prompt et de poser des questions pour le compléter



You

Évalue ce prompt sur une échelle de 1 à 10 (1 le moins bon et 10 le meilleur).
Pose des questions pour compléter ton information.



Je donnerais à ce prompt une note de 8/10. Il est bien structuré et fournit des informations claires. Certaines questions supplémentaires pourraient être posées pour compléter les informations, par exemple :

- Existe-t-il des restrictions spécifiques sur les caractères autorisés pour chaque champ ?
- Le champ "Mots-clés" accepte-t-il des listes séparées par des virgules ou seulement des mots simples ?

Cela permettrait de clarifier complètement toutes les exigences avant de rédiger les scénarios de test.



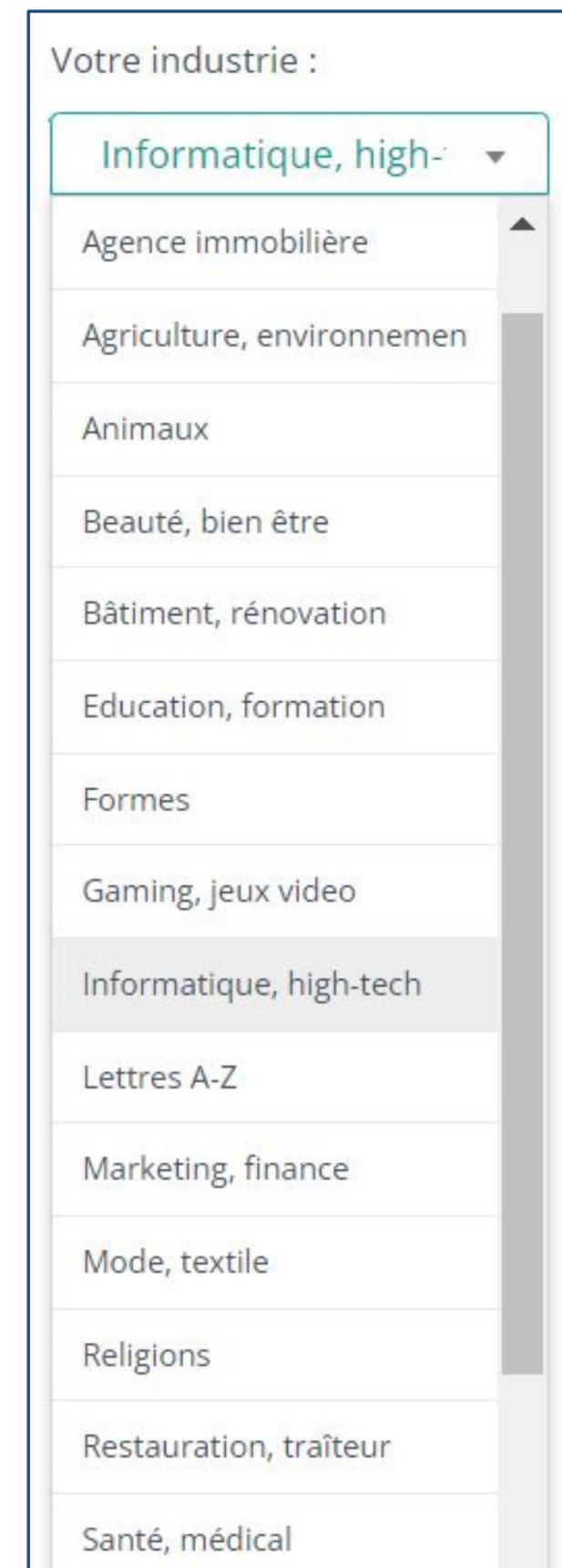
Prompt avec images

Texte + Image : schéma, capture écran, ...

Exemple :

Passer en requête l'image écran d'une liste déroulante, et interroger le modèle sur les valeurs disponibles pour les tests

→ **Données de test**



Prompt avec documents

Texte + Document : normes, spécifications, règles métier, rapports d'anomalies, ...

Exemple :

Passer en document le standard de sécurité informatique ANSSI et réaliser une requête pour établir le plan des tests de sécurité

→ **Planification des tests**

→ **Technique RAG - Retrieval-Augmented Generation - cf. Atelier 3**

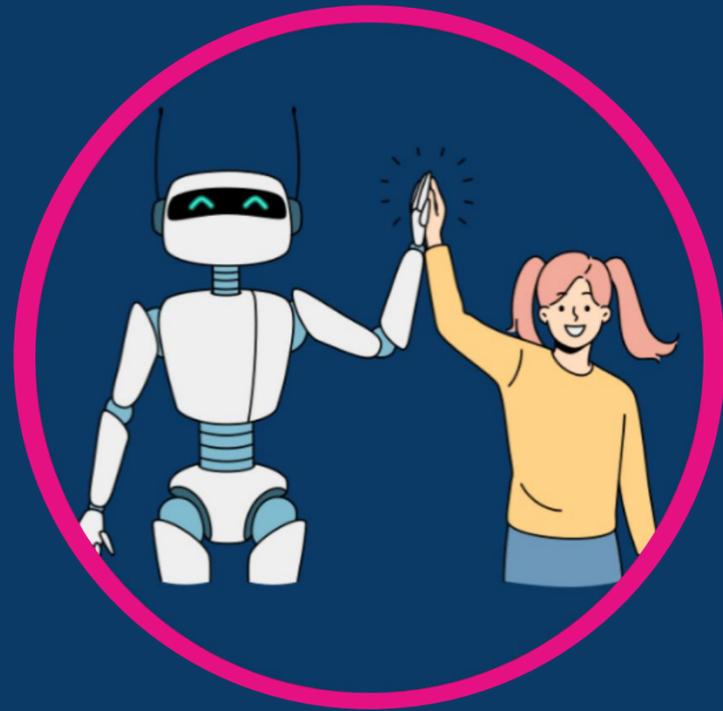


Plan du tutoriel

Requêter une IA conversationnelle pour les tests

- 1 - Introduction : IA générative pour les tests logiciels
 - Atelier 0 - Prise en main du portail LLM
- 2 - Techniques de Prompting
- ● 3 - Ateliers
 - Atelier 1 - Génération de tests
 - Atelier 2 - Priorisation des tests
 - Atelier 3 - Automatisation des tests
- 4 - Bonnes pratiques du prompting pour les tests logiciels
- 5 - Bilan et discussion sur les savoir-faire mis en oeuvre





Tutoriel :

IA pour les tests

Apprendre à prompter

33

Atelier 1
Génération de tests

Mise en contexte de cet atelier



Contexte

- Vous êtes Ingénieur QA au sein d'un éditeur de logiciels dans le domaine de la communication et du marketing. Une application Web de création de logo a été rachetée par votre entreprise. Votre entreprise souhaite réaliser une migration technique vers .NET, qui est la plateforme d'entreprise.
- Il a été constaté qu'aucun test n'avait été fourni lors de la livraison du code : il vous est demandé de développer une suite de tests manuels qui vise à s'assurer de l'absence de régressions lors de la migration technique et de mettre en place une stratégie d'automatisation des tests de régression..

Pourquoi utiliser de l'IA pour vous aider

- Vous avez très peu de temps pour cette tâche et vous souhaitez utiliser un assistant IA pour la création des cas de test et leur mise en forme.





Réalisation de l'atelier

Ce que vous allez faire

- Partie 1 : Conception assistée par étape d'une suite de tests de bout-en-bout et génération du code d'automatisation des tests.
- Partie 2 : Génération de données de tests et variabilisation des tests par LLM utilisant une capture d'écran de l'application.

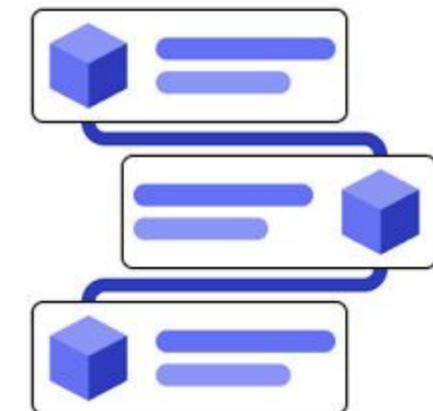
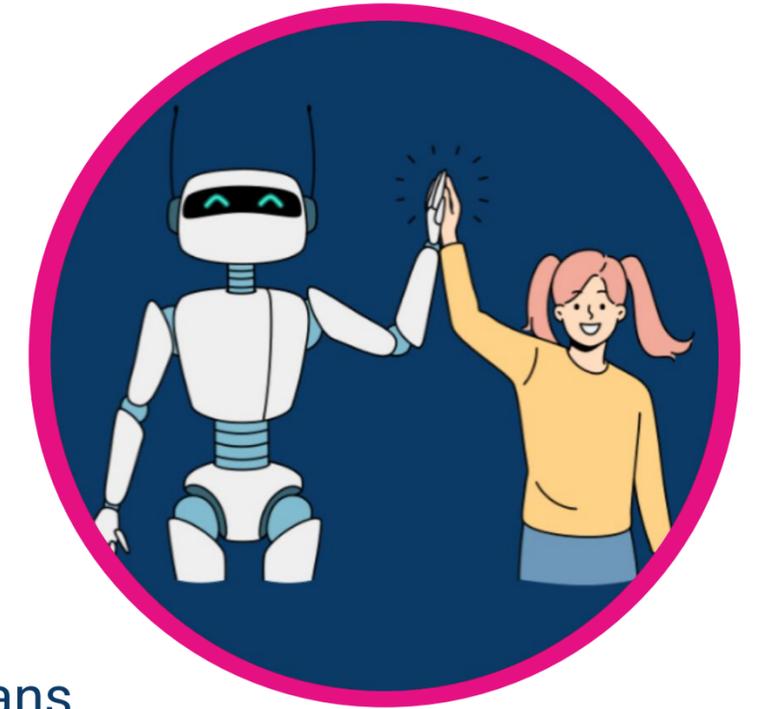
Les consignes sont données sur la page : “JFTL 2024 - Tutoriel IA - Atelier 1” - www.cftl.fr/tuto-ia

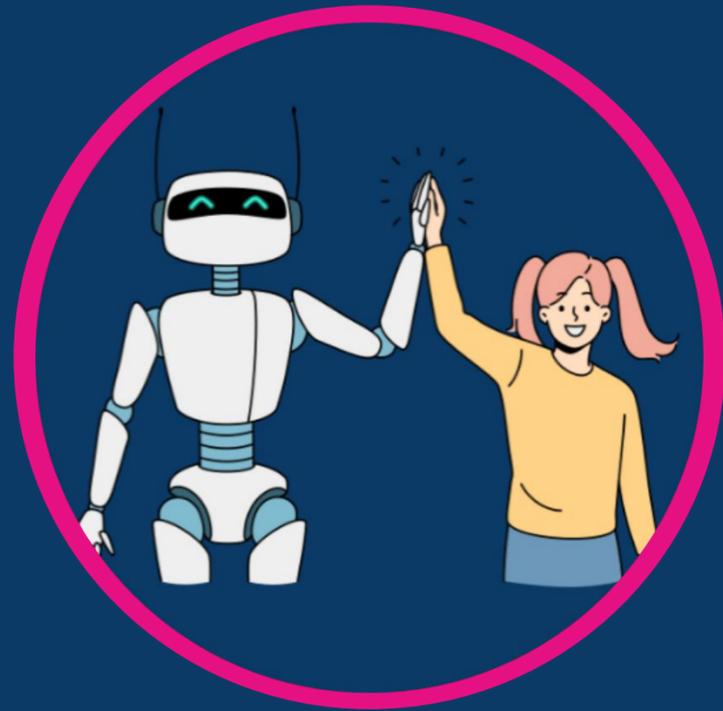


Résumé

Ce que nous avons appris

- A créer des requêtes pour notre assistant IA de conception de tests, produire dans différents formats de sortie (un LLM peut produire du texte, du code, des tableaux, des modèles, ...)
- Techniques de prompting utilisées :
 - Prompt structuré : contexte, rôle, instruction, contraintes
 - Découper une tâche complexe par étapes
 - One-shot / Few-shot prompting : Ajouter un exemple / plusieurs exemples :
- Nécessité d'évaluer le résultat : le résultat de l'IA apporte de l'aide, mais doit être vérifié en détail, et souvent repris / complété





Tutoriel :

IA pour les tests

Apprendre à prompter

37

Atelier 2
Priorisation de tests

Mise en contexte de cet atelier



Contexte

- Vous êtes Ingénieur QA au sein d'un éditeur de plateforme d'enseignement pour les universités.
- Les cas de tests manuels doivent être ordonnancés avant exécution car ils y a des dépendances à prendre en compte et des priorités du fait que probablement tous les tests ne pourront pas être exécutés par manque de temps.

Pourquoi utiliser de l'IA pour vous aider

- L'ordonnancement des campagnes de test est souvent compliquée à réaliser. Vous vous dites que l'apport de l'IA serait un gain de temps et d'effort.





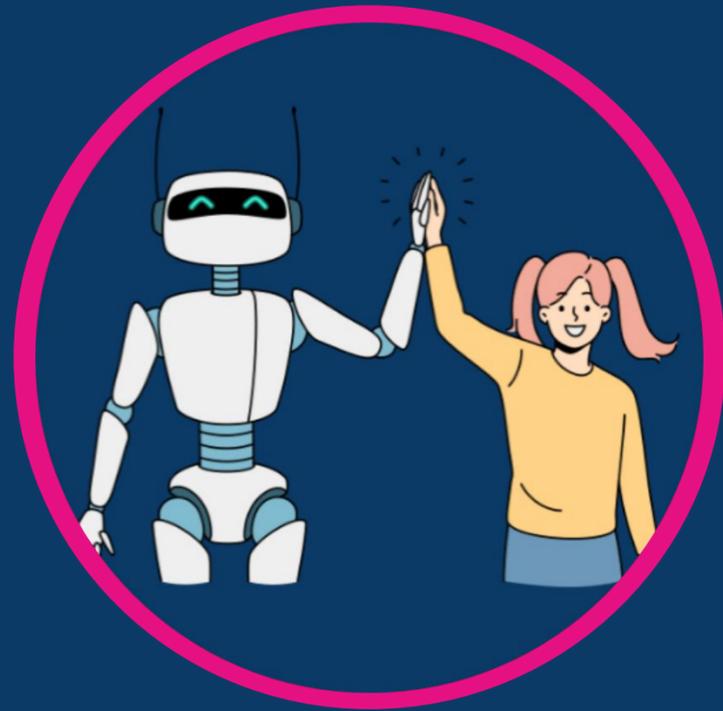
Réalisation de l'atelier

Ce que vous allez faire

- Utiliser l'IA générative pour une tâche d'ordonnancement de l'exécution des tests
 - Chercher à améliorer votre prompt
- et aussi...
- Analyser les résultats sur une tâche nécessitant des capacités de raisonnement

Les consignes sont données sur la page : “JFTL 2024 - Tutoriel IA - Atelier 2” - www.cftl.fr/tuto-ia





Tutoriel :

IA pour les tests

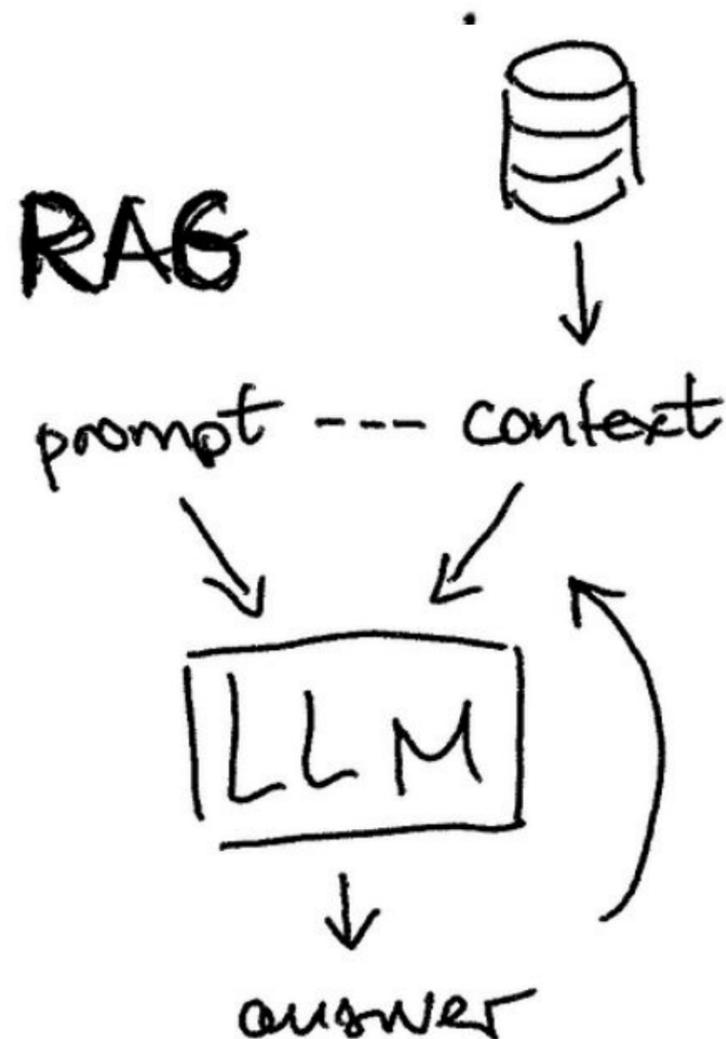
Apprendre à prompter

Atelier 3
Technique RAG

Retrieval-
Augmented
Generation

Technique RAG - Présentation

Augmenter le contexte de la requête par l'extraction de sources documentaires



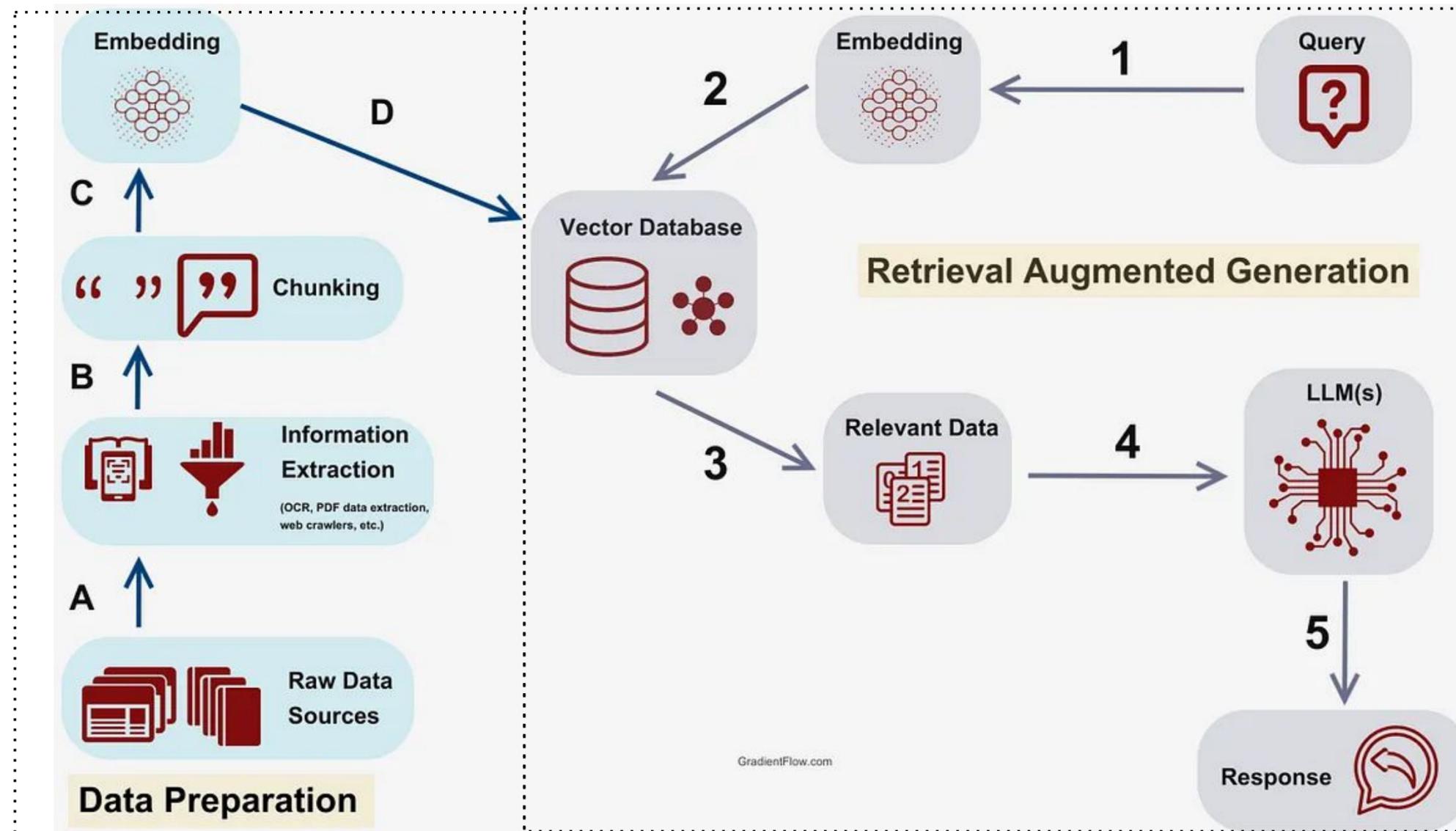
RAG :
Retrieval-
Augmented
Generation

Donner du contexte spécifique au projet / à l'organisation lors d'une tâche à réaliser via LLM



La technique RAG en un schéma

Phase 1 :
Préparation
des
données



Phase 2 :
Utilisation
lors d'une
requête à
l'IA





Réalisation de l'atelier

Ce que vous allez faire

- Partie 1 : Questions / Réponses avec RAG sur un document complexe (Recommandations de sécurité ANSSI pour l'IA générative)
- Partie 2 : Analyse de tests à partir d'un document de spécifications techniques d'une application de turbine éolienne (petit exemple pédagogique) avec RAG

Les consignes sont données sur la page : “JFTL 2024 - Tutoriel IA - Atelier 3” - www.cftl.fr/tuto-ia



Avantages et points d'attention du RAG

Application RAG dans mon organisation de test

Avantages

- Exploitation de données de l'entreprise
- Amélioration de l'exactitude des faits et réduction des hallucinations
- Transparence et explication

Points d'attention

- Complexité accrue de la chaîne LLM et latence potentielle
- Dépendance à l'égard de la qualité des données
- Réglage des hyperparamètres



Plan du tutoriel

Requêter une IA conversationnelle pour les tests

- 1 - Introduction : IA générative pour les tests logiciels
 - Atelier 0 - Prise en main du portail LLM
- 2 - Techniques de Prompting
- 3 - Ateliers
 - Atelier 1 - Génération de tests
 - Atelier 2 - Priorisation des tests
 - Atelier 3 - Automatisation des tests
- ● 4 - Bonnes pratiques du prompting pour les tests logiciels
- 5 - Bilan et discussion sur les savoir-faire mis en oeuvre



4- Bonnes pratiques du prompting

Prompt engineering adapté aux activités des tests logiciels

Des objectifs clairs et spécifiques

- Préciser le contexte
- Définir le résultat attendu

Des données structurées et détaillées

- Formats définis (csv, xml, json, ...)
- Exemples donnés - few-shot prompting

Raffinement itératif du prompt

- Commencer général puis préciser
- Utiliser l'auto-vérification, le méta-prompting, et la vérification humaine

Utiliser un style précis et intégrer le multimodal

- Définir une terminologie et s'y tenir
- Intégrer des images écran (suivant LLM)



Le prompt “système”

Prompt avant le prompt

MISTRAL-MEDIUM-LATEST

Context window: 32768 tokens

Used: 290 tokens

Tu es mon assistant pour la conception de tests logiciels. Tu dois me répondre uniquement en français.

[Hide system prompt](#)



Outiller la création des prompts

Pour travailler en équipe, capitaliser, évaluer, réutiliser, ...

Utiliser / créer des bibliothèques de template de prompts

- Pour votre organisation et vos activités récurrentes
- Smartesting développe une bibliothèque de prompts
- Avec l'expérience : développez votre bibliothèque interne

Utiliser un IDE de Prompt Engineering

- Domaine en évolution
- Ces IDE sont destinés à faciliter la création de prompts lorsque les tâches sont très variées.
- Ex. OpenAI Playground et Anthropic Workbench



Quelques ressources

Bases du Prompt Engineering

- www.promptingguide.ai
 - Introduction au prompting
- Taxonomie des techniques de prompting
 - <https://arxiv.org/pdf/2402.07927.pdf>

Aggrégateurs LLM

- POE - <https://poe.com/>
- Perplexity - <https://www.perplexity.ai>

Utiliser un IDE de Prompt Engineering

- <https://platform.openai.com/playground>
 - L'interface de mise au point des Prompts GPT-x lorsque utilisé en mode API
- <https://console.anthropic.com/workbench>
 - L'interface de mise au points de Prompts de Anthropic
- <https://promptmetheus.com/>
 - Permet d'itérer et de qualifier vos prompts



Stratégie / Feuille de route

Comment en pratique ?

WEBINAIRE

IA GÉNÉRATIVE POUR LES TESTS :
STRATÉGIE ET FEUILLE DE ROUTE


BRUNO LEGEARD

 JEUDI 27 JUIN
 12h15 (Heure Française)

Inscrivez-vous  

 x  **UpSkill4IT**
E-LEARNING & E-COACHING



5- Bilan et discussion

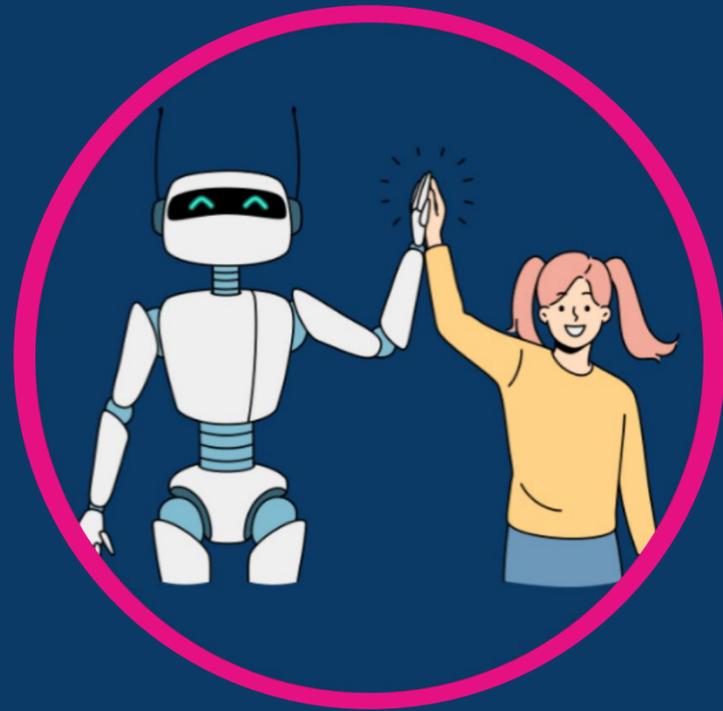
Ce que nous avons appris

- La **signification et l'importance** du prompt engineering
- La **structure de base** d'un prompt
- Les **techniques** les plus utiles du prompting
- Les **bonnes pratiques** du prompting
- L'ampleur des **techniques avancées**

Ce que nous avons pratiqué

- Le **prompting** avec exemples
- Le requêtage **structuré**
- Le requêtage **par étapes**
- Le requêtage **non-structuré en cascade**
- La **comparaison** de LLM
- La génération de différents **artefacts de tests**





Tutoriel :

IA pour les tests

Apprendre à prompter

- Bruno Legeard
- Fabrice Bouquet
- Abbas Ahmad
- Anne Kramer
- Alexandre Vernotte