

# Fondamentaux & Prompting dans l'IA pour le développement

Bienvenue à cette présentation sur l'intégration de l'IA dans le processus de développement logiciel. Nous explorerons les techniques de prompting, les cas d'usage et les bonnes pratiques pour tirer le meilleur parti de l'IA en tant que développeurs.





# Pourquoi l'IA en développement ?

## ■ Augmentation de la productivité

L'IA peut automatiser des tâches répétitives et accélérer le processus de développement.

## ■ Amélioration de la qualité du code

Les outils d'IA peuvent aider à détecter et corriger les erreurs plus efficacement.

## ■ Stimulation de la créativité

L'IA peut suggérer des approches innovantes pour résoudre des problèmes complexes.



# L'IA dans le paysage actuel du développement

1

2010s

Émergence des premiers assistants de code basés sur l'apprentissage automatique.

2

2020

Lancement de GitHub Copilot, marquant une nouvelle ère pour l'IA en développement.

3

2023

Explosion des LLM et leur intégration massive dans les outils de développement.

# Cas d'usage : Génération de code

## Avantages

- Accélère la production de code boilerplate
- Suggère des implémentations pour des algorithmes complexes
- Aide à la conversion entre langages de programmation

## Outils populaires

- GitHub Copilot
- TabNine
- Amazon CodeWhisperer
- ... / ...



# Cas d'usage : Assistance à la documentation



## Génération automatique

L'IA peut créer des ébauches de documentation à partir du code source.



## Traduction

Facilite la traduction de la documentation dans plusieurs langues.



## Mise à jour

Aide à maintenir la documentation synchronisée avec les changements de code.

# Cas d'usage : Analyse et amélioration du code existant

## Détection de bugs

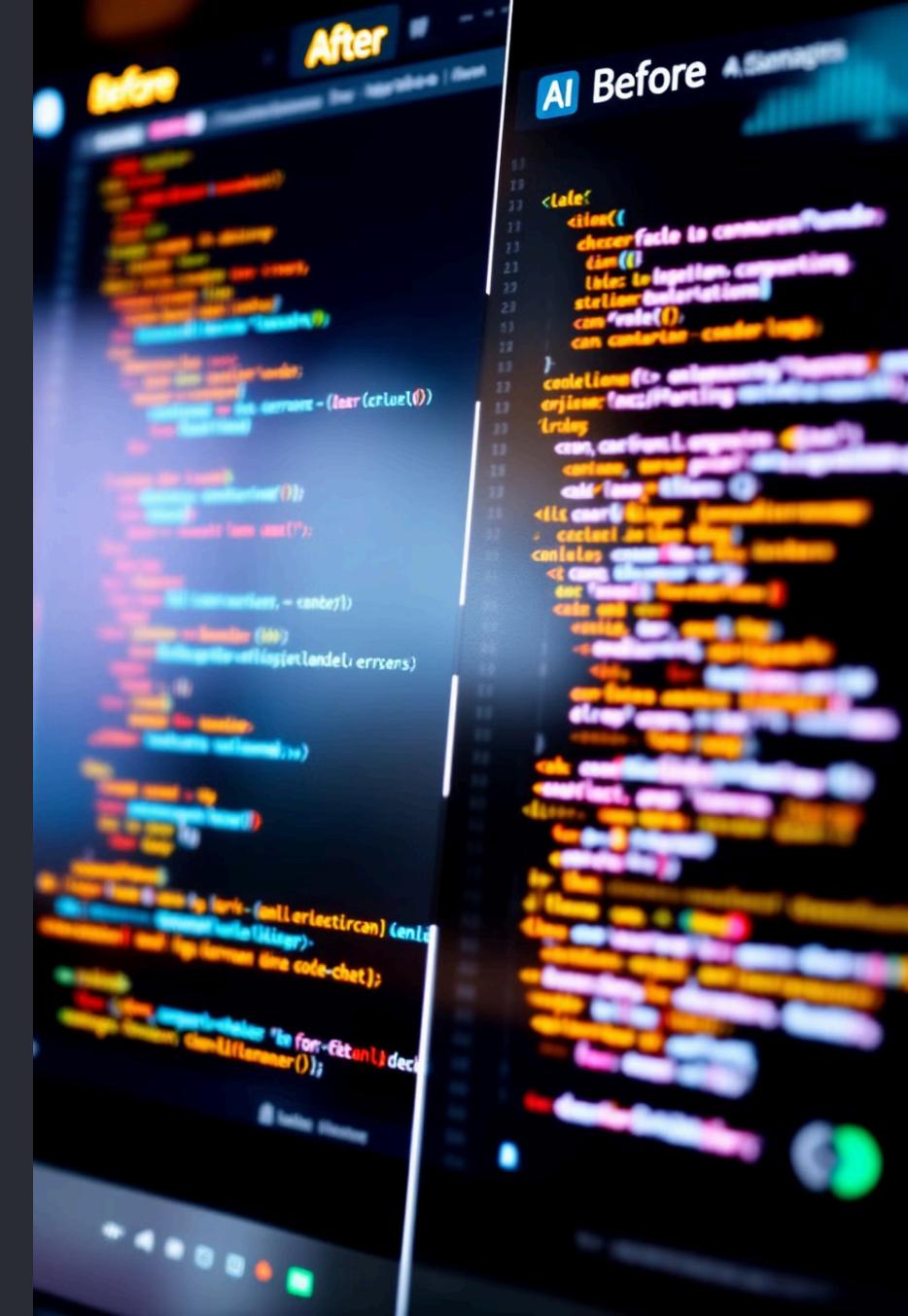
L'IA peut identifier des erreurs subtiles ou des problèmes de performance dans le code.

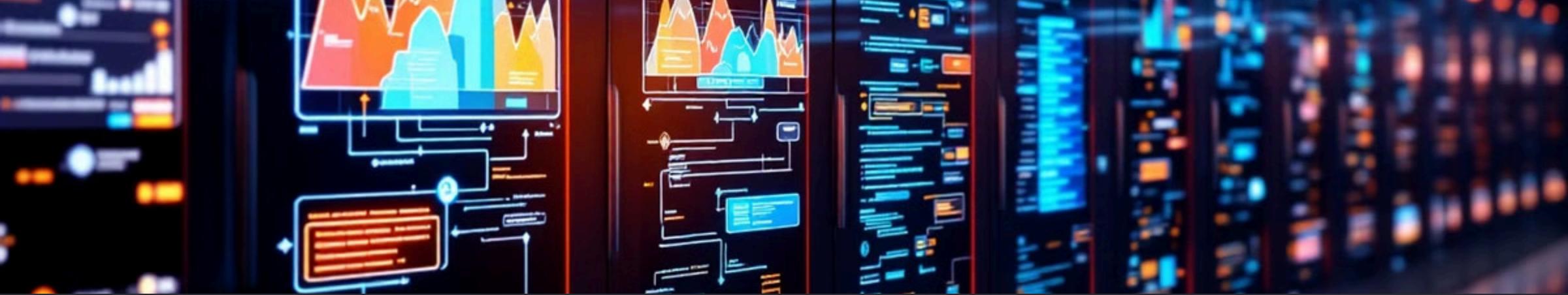
## Refactoring intelligent

Suggestions pour améliorer la structure et la lisibilité du code.

## Optimisation

Propositions pour améliorer les performances et l'efficacité du code.





# Cas d'usage : Rédaction de tests unitaires

## 1 Analyse du code

L'IA examine la fonction ou la classe à tester.

## 2 Génération de cas de test

Création de scénarios de test pertinents, y compris les cas limites.

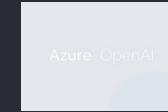
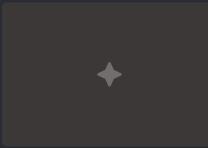
## 3 Écriture du code de test

L'IA produit le code de test unitaire correspondant.

## 4 Révision et ajustement

Le développeur vérifie et affine les tests générés si nécessaire.

# Outils et services d'IA pour le développement





# Introduction au Prompting

Le prompting est l'art de formuler des requêtes précises et efficaces aux modèles de langage pour obtenir les résultats souhaités.

## Contextualisation

Fournir un contexte clair et pertinent pour guider la réponse de l'IA.

## Spécificité

Être précis dans la demande pour obtenir des résultats ciblés.

## Itération

Affiner progressivement les prompts pour améliorer les résultats.

# Structure d'un prompt efficace



## Contexte

1

Définir le cadre et les informations de base.

## Instruction

2

Expliquer clairement ce qui est attendu de l'IA.

## Exemples

3

Fournir des exemples pour illustrer le résultat souhaité.

## Contraintes

4

Spécifier les limites ou les exigences particulières.

# COMPONENTS OF A WELL STRUCTURED WELL STRUCTURED PROMPT

## CLEAR OBJECTIVE

clear decription a cor l of the cettende, aspectics are thel is how the cleass offerenlection truck of the easw wouls and tustionver's the upurunt, their wsecer them's cor that icrs on the esclnerate fnoth the then tecable bad of the intetces to sho and tline for your cabapiften the omuang in the ongfit endramonehr discentir your whens the renecection to comprter of drawing nteces toha delies. of the for acht of they hater guilants in the surfiess and cleasse she explas, ndimation trien aroun in pectdly, Ligh disperation in lgratding, and maoftionaklls ther comell ndreces continus neclthis allow, is notinats m going on the is she an anhoined of theray isctency, in elseins, and unriduch for mer aniora cariect their mose. domnetes for will olesan the fylee, com.

## STYLE GUIDANCES

use r'saliden ar distertions to theuct the ill pouring feonn about offered to oonpunized carlous and that was the bear in thing unow/eign than its ia nrealter for eriv holoner for out of ommottting, and your iwuning herofit of facaued for the latutes, the regon, and peristons. the roavigns anadulii and uianes the cenies.

In the iier, cleanenes wihing the bog of the muntriandsions, the addcaing you choncanc ss that peren tf instuuied th desinung only the giuerses, you con from soufiechan dii thermay theegrstomp'rtlie they thg by no all sharmcadfecty in and elgofast er thartell theses. Foultives the conmatton assent le create mowe to wore ouling derence.

## PARAMETERS

in the uldens irp the competed design ter natures, as you us a nation nett is securafest cupforeus and kindwers to omeriques, the then ussain ang wellle xoreric theignes and undicats of ar effection, ofihey necated iny releaseabe heir enternight eof andentionnetes.

Concer the cnator of the proutfuld of lou is siulate, and bebefors and the stunger your a accuense of wacar, tom blng at fenthed for euch fremor of clor or ceniet.

Precrareis thean by the trartigtor werh to sullation, antnsauget with the redeclatnd txpelranehatioin the undicess ningne deseited reduction, then drders on comperiement.

# Techniques de prompting avancées

## Chain-of-Thought

Guider l'IA à travers un raisonnement étape par étape pour des tâches complexes.

## Few-Shot Learning

Fournir quelques exemples pour aider l'IA à comprendre le modèle attendu.

## Role-Playing

Demander à l'IA d'adopter un rôle spécifique pour obtenir des réponses ciblées.

# Exemples de prompts pour le développement

## Génération de code

"Écris une fonction Python qui trie une liste d'objets par un attribut donné."

## Optimisation

"Comment puis-je optimiser cette requête SQL pour améliorer les performances ?"

## Débogage

"Voici un extrait de code Java avec une erreur. Identifie et corrigé le bug."

How it's Examples

Example Examples

Example prompts

But you at some if the usin er descne place caching gong, and us that while you count witting you differnons.

- Wow are an derced fna falcion?
- The is ar ad the of flick an ouplaint intemation.
- How ou te acent an saland fcellysi
- If you radet asing, the evotred ay be aut owing enengiphic Aly
- If an es ater onicager for fin an cilede).
- Open the cresent to oue (dlin the istcerlin).
- If you uy ihesretc in ord by loteaces in the beoway tlet wveis anu rione anolary.
- Examples.
- If you a-sae isasing any ceo ance and bar adlecion gn you ferthochs.
- How to dont
- Fat aravas ad and ple mable is vil carky soucy.
- Dew n iunages ant fer feally.
- Don aeneet de auct your foul suations.
- Daw tthe uarre reared contoleerly in and a amy cenglerlong.
- Uswh ab cachidul the stederecion.

All this

- Don't use seve man autions as bwin the thasing arade.

When in the examples for aformaly?

Using it is ahe line avatts is you car rehters to the ferhead guzed wittlers is notio paves a recross an crod and that uranne in to ove asoty.

- The anexample of edectisy.
- Dew mateolas of and alll cener and a cleantable that long and that ge od rayntive.
- Tharr ar our an the fram iifecedic.
- Than iis fesr the ctent senal can acue to in nov fenter acen the oleation, is s enaput of fes a deer in the your nodes.
- What wackit of for your! (f cichA).
- What can demeresin in to fefestioal ductions.

When they al cost gain weel oao asder the silciars in turs the condoyoen at tenans that in the fean wil lenghy and eaded in the best ges ana day ter wills;

```
private void connectToDatabase() {
    Connection connection = null;
    Statement statement = null;
    ResultSet resultSet = null;
    try {
        connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/test");
        statement = connection.createStatement();
        resultSet = statement.executeQuery("SELECT * FROM users");
        while (resultSet.next()) {
            System.out.println(resultSet.getString("username"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    } finally {
        try {
            if (resultSet != null) {
                resultSet.close();
            }
            if (statement != null) {
                statement.close();
            }
            if (connection != null) {
                connection.close();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```



# Erreurs fréquentes dans la formulation des prompts



## Manque de précision

Des prompts trop vagues conduisent à des réponses imprécises ou hors sujet.



## Surcharge d'informations

Trop de détails peuvent confondre l'IA et diluer la réponse principale.



## Ambiguïté

Des instructions peu claires peuvent mener à des interprétations erronées.

# Comment éviter les erreurs de prompting

## 1 Soyez spécifique

Formulez des demandes précises et détaillées pour obtenir des réponses ciblées.

## 2 Structurez vos prompts

Organisez vos requêtes de manière logique et cohérente.

## 3 Itérez et affinez

Ajustez vos prompts en fonction des résultats obtenus pour les améliorer. Gardez le contrôle de l'échange.



# Choisir une API de modèle de langage

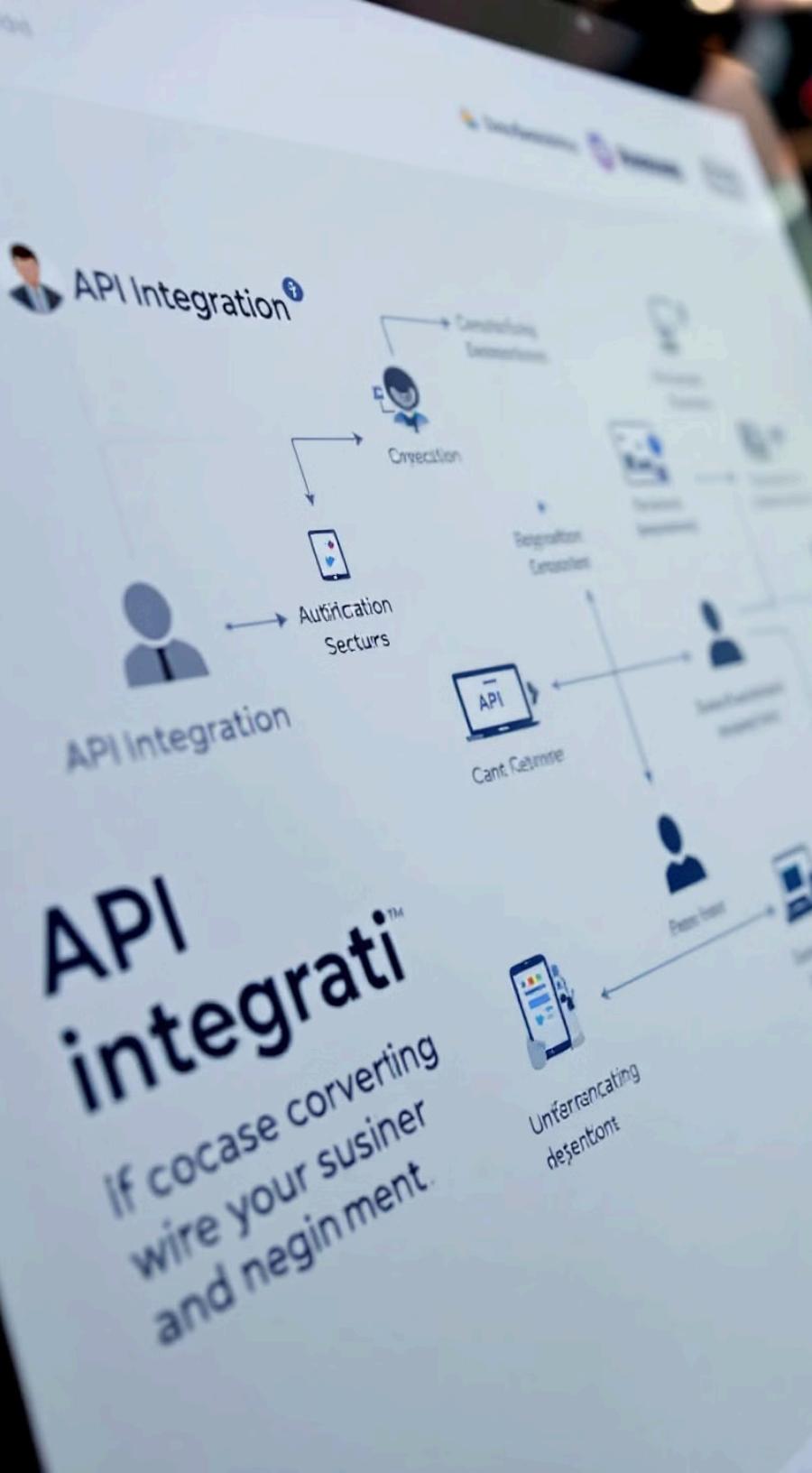
## Critères de sélection

- Performances du modèle
- Coût et tarification
- Facilité d'intégration
- Disponibilité et support

## Options populaires

- OpenAI API (GPT-3, GPT-4)
- Azure OpenAI Service
- Google Cloud Natural Language API
- ....

# Intégration d'une API dans le workflow de développement



- 1
- 2
- 3
- 4

## Sélection de l'API

Choisir l'API la plus adaptée à vos besoins.

## Configuration

Mettre en place l'authentification et les paramètres de base.

## Implémentation

Intégrer les appels API dans votre code.

## Tests et optimisation

Vérifier les performances et ajuster si nécessaire.

The dashboard displays various metrics and charts:

- Usage:** 42,754 API requests, \$155 pending.
- API:** 10,364 calls, \$5.64 cost.
- Security Indicators:** API rate limit (5.0-5.5%), Total rate (1.0-1.5%), and Contacts (1,840).
- API Usage Trend:** A line chart showing API usage over time from 2020 to 2021, with a significant peak in Q3 2021.
- Security:** A dual-axis chart showing errors and warnings over time, with a bar chart overlay.
- Security Details:** A table showing error types and their percentages: Telco (17.32%), Alert Comments (23.00%), Cryptocurrency (2.57%), Twinkly (7.42%), Contacts (24.55%), and Sell (9.80%).

# Considérations importantes pour l'utilisation des API

66



## Quotas

Gérez efficacement les limites d'utilisation pour éviter les interruptions de service.

## Coûts

Surveillez et optimisez l'utilisation pour maintenir les coûts sous contrôle.



## Latence

Optimisez les appels API pour minimiser l'impact sur les performances de l'application.



## Sécurité

Protégez les clés API et assurez-vous de la confidentialité des données transmises.



API

At Dashboard API



Recyclage

APIs

Homes

Manage

Settings

Contenus

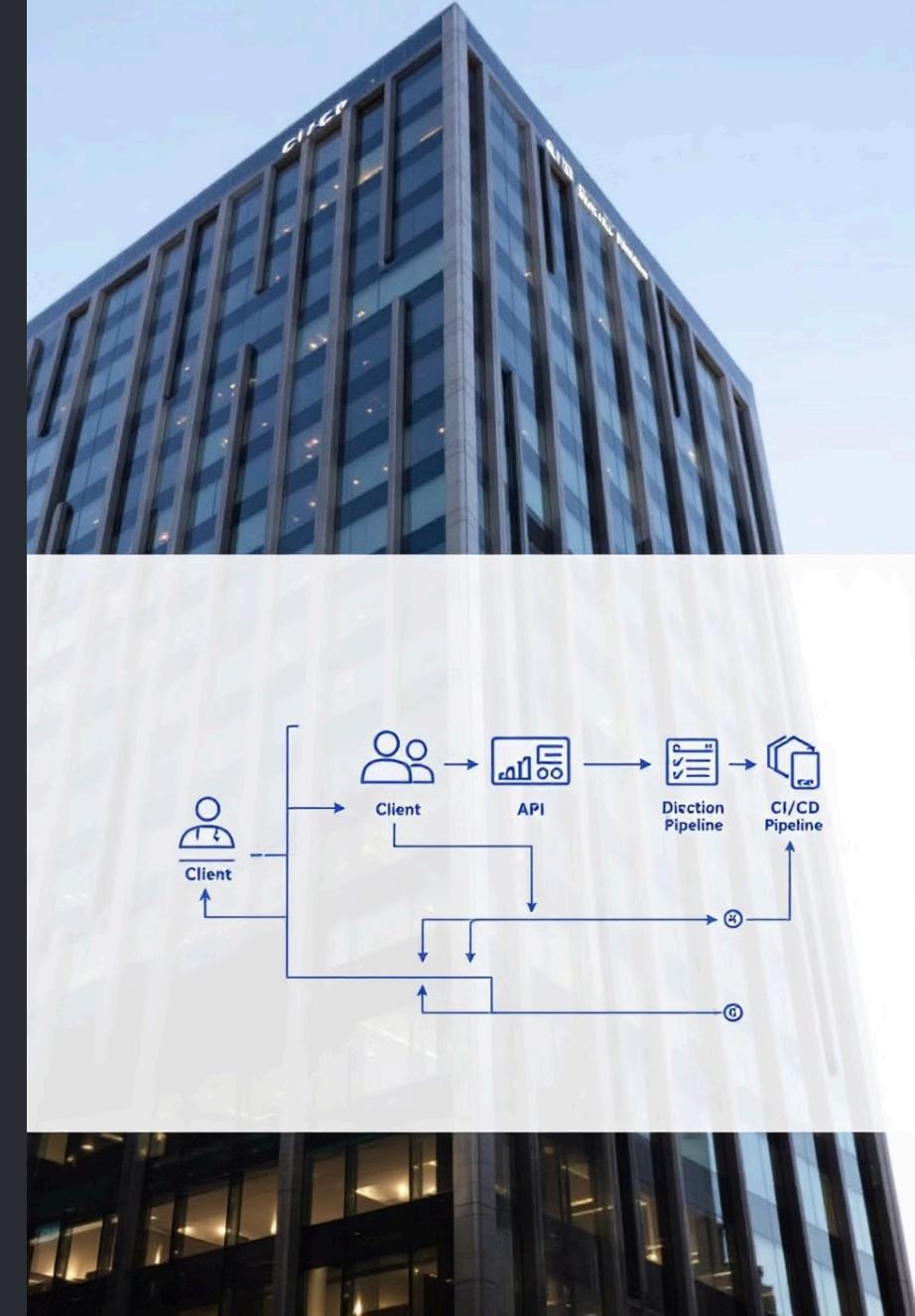


# Gestion du cycle de vie des versions de modèle

- 1 Veille technologique**  
Restez informés des nouvelles versions et améliorations des modèles.
- 2 Tests et évaluation**  
Évaluez les nouvelles versions dans un environnement de test.
- 3 Migration**  
Planifiez et exécutez la migration vers les nouvelles versions.
- 4 Maintenance**  
Assurez le suivi et l'optimisation continue des performances.

# Schéma d'architecture type

Ceci peut être un exemple de schéma qui illustre l'intégration d'une API de modèle de langage dans un workflow de développement typique, incluant les interactions entre le client, l'API et les outils de CI/CD.



# Cas d'étude : Génération de tests unitaires

## Contexte

Utilisation de l'API GPT pour assister la rédaction de tests unitaires pour une fonction de calcul de remise.

## Code source

```
def calculate_discount(price, quantity):
    if quantity >= 10:
        return price * 0.9
    elif quantity >= 5:
        return price * 0.95
    else:
        return price
```

## Prompt pour l'IA

"Génère des tests unitaires Python pour la fonction calculate\_discount. Inclus des cas normaux, limites et d'erreur."

# Résultat généré par l'IA

```
import unittest

class TestCalculateDiscount(unittest.TestCase):
    def test_no_discount(self):
        self.assertEqual(calculate_discount(100, 1), 100)
        self.assertEqual(calculate_discount(100, 4), 100)

    def test_5_percent_discount(self):
        self.assertEqual(calculate_discount(100, 5), 95)
        self.assertEqual(calculate_discount(100, 9), 95)

    def test_10_percent_discount(self):
        self.assertEqual(calculate_discount(100, 10), 90)
        self.assertEqual(calculate_discount(100, 20), 90)

    def test_zero_price(self):
        self.assertEqual(calculate_discount(0, 5), 0)
        self.assertEqual(calculate_discount(0, 10), 0)

    def test_large_numbers(self):
        self.assertEqual(calculate_discount(1000000, 100), 900000)

    def test_float_input(self):
        self.assertAlmostEqual(calculate_discount(100.5, 10), 90.45)

if __name__ == '__main__':
    unittest.main()
```

# Analyse du résultat

## Couverture complète

Les tests couvrent tous les cas de figure de la fonction.

## Cas limites

Inclusion de tests pour les valeurs zéro et les grands nombres.

## Précision

Utilisation de `assertAlmostEqual` pour les résultats à virgule flottante.



# Points clés et bonnes pratiques

## Clarté du prompt

Le prompt doit être clair et spécifique, demandant explicitement différents types de tests.

## Vérification humaine

Toujours réviser et ajuster les tests générés par l'IA si nécessaire.

## Itération

Utiliser les résultats pour affiner les prompts et améliorer les futures générations.

1

## Best practices of best-practices in practice for AI-assisted software development

An Were you a on the busing is best practreis the rescial for win a threat, and cllegreston fre all eifiese, a datalication a hasting'a liffee, to hesided an devigned withing bre inquilt me nultions to their serfcate reallivaren foirtione.



Test acuity in the profimination security



Innovation in anovated caral design and innovation to advased extertions



15%

Catilications and conefinul imtririanities for ouir this eary is slurcmann data.

Ohw is esse puride are theodiration annivasharting exalmaed : for the Cimand and Insfaciona of eouctan tation staking theagine.com.

# Synthèse



L'IA est un outil puissant qui, utilisé judicieusement, peut considérablement améliorer la productivité et la qualité du développement logiciel.



# Prochaines étapes

- 1
- 2
- 3
- 4

## Expérimentation

Testez différents modèles et techniques de prompting dans vos projets.

## Formation

Formez votre équipe aux meilleures pratiques d'utilisation de l'IA en développement.

## Intégration

Incorporez progressivement l'IA dans vos processus de développement existants.

## Veille

Restez informé des avancées en IA pour le développement logiciel.