

Renforcez vos
compétences en
codage IA/ML avec le

Cours de Codage MLOps

Ambassadeur MLflow - 5 juin 2024

Médéric HURIER et Matthieu JIMENEZ

Qui sommes-nous ?



Dr. Médéric HURIER

Ingénieur MLOps freelance

Travailler pour [Decathlon Digital](#)

Doctorat en Sécurité informatique et IA

<https://www.fmind.dev/>



Dr. Matthieu JIMENEZ

Chercheur associé

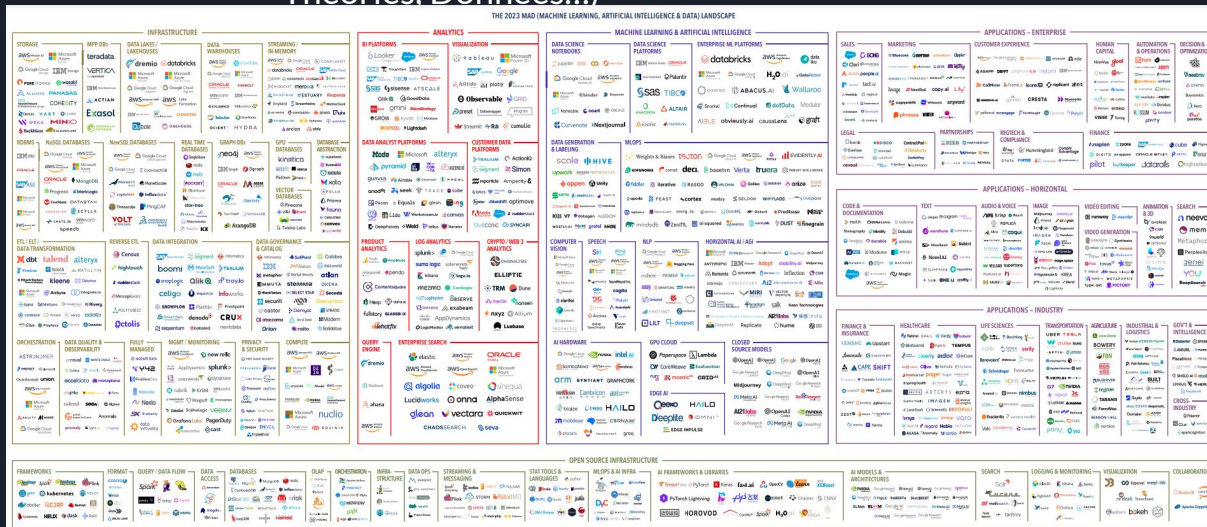
Travailler pour [Université du Luxembourg](#)

Doctorat en Sécurité informatique et IA

<https://website.jimenez.lu/>

Par où commencer ?

Beaucoup de choses à savoir en Machine Learning (Frameworks, Solutions, Théories, Données...)



Paysage dément 2023

Mais qu'en est-il de ceux liés au code ?

De nombreux concepts et technologies du génie logiciel s'appliquent au ML, en particulier en production.

Une solution est de demander à un ingénieur ML, mais cela entraîne un goulot d'étranglement et est entravé par des problèmes de communication.



Équipe
d'ingénierie ML

Science
des
données
équipe

Médéric & Matthieu

Package Python MLOps

Là où tout a vraiment commencé

Créé à partir d'un modèle MLOps réel

Inclure de nombreuses bibliothèques open source utiles pour le MLOps.

Modèles de conception intégrés nativement.

Problème : comment expliquer toutes les notions sous-jacentes ?



© datascientest

<https://github.com/fmind/mlops-python-package>

L'idée

Créer un cours pour combler le fossé des connaissances

Devrait être **agnostique de la plateforme**
Il existe déjà de nombreux cours sur les solutions éditeurs.

Conçu pour être applicable immédiatement
(par exemple, pour les collègues de Mederic chez Decathlon)



© wikicommons

Présentation du MLOps Coding Course

Publié le 22 mai 2024

7 chapitres couvrant divers sujets MLOps du point de vue du code

Les chapitres suivent le chemin naturel d'un projet

Organisé autour d'un système de questions-réponses, avec des ressources supplémentaires et des exemples

Couplé au package Python MLOps

MLOps Coding Course



```
33 KIND: T.Literal["TrainingJob"] = "TrainingJob"
34
35 # Run
36 run_config: services.MlflowRunConfig = services.MlflowRunConfig(name="Training")
37 # Data
38 inputs: datasets.ReaderK
39 targets: datasets.ReaderK
40 # Model
41 model: models.ModelKind = models.ModelKind(name="KIND")
42 # Metrics
43 metrics: list[metrics.Metric] = list[metrics.Metric](discriminator="KIND")
44 # Splitter
45 splitter: splitters.Splitter = splitters.Splitter(discriminator="KIND")
46 # Saver
47 saver: registries.Saver = registries.Saver(discriminator="KIND")
48 # Signer
49 signer: signers.SignerKind = signers.SignerKind(name="KIND")
50 # Registry
51 # - avoid shadowing pydantic
52 registry: registries.RegistryKind = registries.RegistryKind(name="KIND")
53
```

<https://mlops-coding-course.fmind.dev/>

Place au cours !



Overview

Course, Project,
Plateforme



Initializing

Python, pip, pyenv,
Poetry, git, Vscode



Prototyping

Notebook, Import,
Config, Dataset, Analysis,
Modeling



Productionizing

Package, Module,
Paradigm, Entrypoint,
Configuration,
Documentation



Validating

Typing, Linting, Testing,
Logging, Security,
Debugging



Refining

Design Pattern, Task
Automation, CI/CD,
Container, Model Registry



Sharing

Repository, License,
Readme, Release,
Template

0. Aperçu: une introduction au cours

Indexation à base zéro 

- 0.0. Cours De quoi parle ce cours ?
- 0.1. Projets Par défaut ou apportez le vôtre !
- 0.2. Jeux de données Sur la (non-)structure des données
- 0.3. Plateformes Nous sommes agnostiques aux plateformes
- 0.4. Mentorat Séances d'apprentissage individuel
- 0.5. Assistants Un RAG premium pour vous
- 0.6. Ressources Aller au-delà du cours



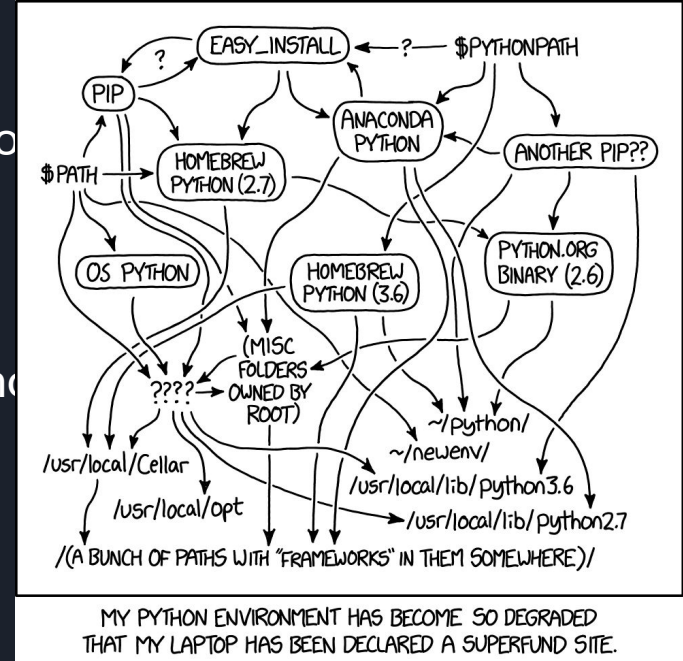
Things You Learn

<https://xkcd.com/1775/>

Initialisation Préparez votre environnement de dev.

Apportons nos Pythons 🦆

- 1.0. Système Vous avez mon système d'exploitation
- 1.1. Python: Et mon Interpréteur Python
- 1.2. Pyenv: Et mon environnement Python
- 1.3. Poetry: Et mon gestionnaire de paquets Python
- 1.4. Git Et mon Système de contrôle de version
- 1.5. GitHub Et mon dépôt de code
- 1.6. VS Code Et mon éditeur de code

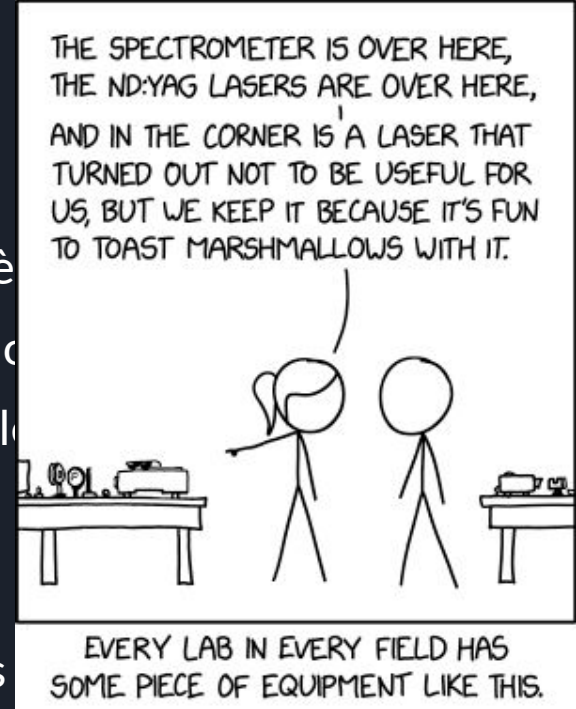


Python Environment
<https://xkcd.com/1987/>

2. Prototypage Explorez votre espace de recherche

Vers les Jupyter notebooks et au-delà 🪐

- 2.0. Notebooks Réserver ou ne pas réserver ?
- 2.1. Importations: Grâce à la puissance de l'écosystème
- 2.2. Configurations Séparer le code de la configuration
- 2.3. Jeux de données: Comment charger et manipuler les données
- 2.4. Analyse: Apprendre à faire parler les données
- 2.5. Modélisation Créer des modèles d'IA/ML et des
- 2.6. Évaluations Mettre la science dans la modélisation

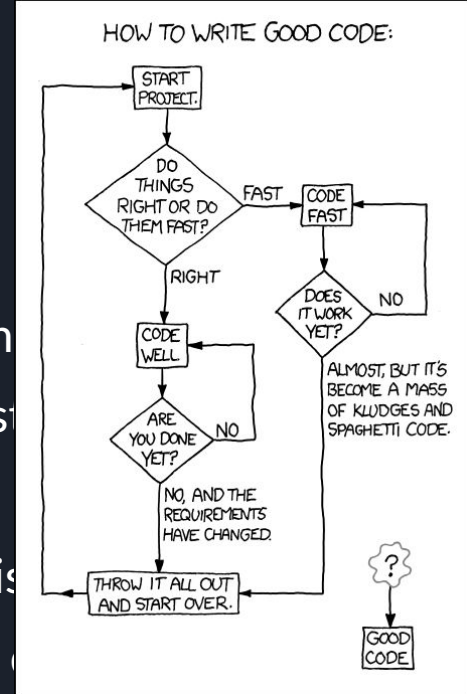


Équipement de laboratoire
<https://xkcd.com/2514/>

3. Mise en production: Allez, atteignez votre public.

Entrée dans l'Ère Industrielle ⚙️

- 3.0. Paquet: Paquet = Métadonnées + Code
- 3.1. Modules Diviser en modules et conquérir
- 3.2. Paradigmes La meilleure philosophie de programmation
- 3.3. Points d'entrée Exposer des paquets à d'autres systèmes
- 3.4. Configurations Éditez vos configs, pas votre code
- 3.5. Documentations De bonnes docs font de bons amis
- 3.6. Espace de travail VS Code Organisez votre espace de travail



Bon code

<https://xkcd.com/844/>



3. Mise en production - Points clés

- Construire un package Python est la clé.
- La modularité contribue à séparer les préoccupations.
- Séparer les configurations du code
- Exposez votre solution à d'autres systèmes

4.0 Validation: assurance & réassurance

Pas de trace de méthodes formelles 😊

- 4.0. Typage Les types sont des schémas
- 4.1. Linting Les linters sont des garde-fous
- 4.2. Test Les tests sont des spécifications exécutables
- 4.3. Journalisation La journalisation est votre service de renseignement
- 4.4. Sécurité La sécurité est la solution, le garde du corps
- 4.5. Formatage Codez avec style grâce aux outils de formatage
- 4.6. Débogage Le débogage est le meilleur tueur de bugs



Monde académique vs. Monde de l'entreprise

<https://xkcd.com/664/>



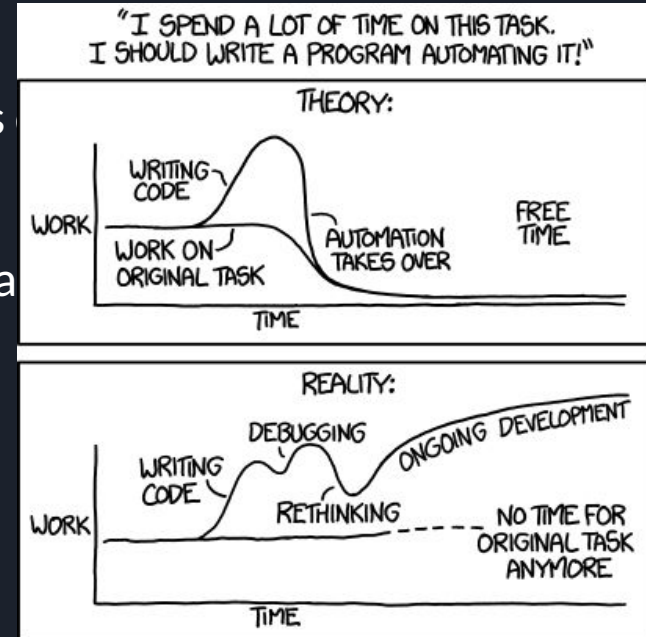
4.0 Validation - Points clés

- La validation logicielle est un investissement...
- Mais cela a des retours positifs sur le long terme.
- Vous ne voulez pas vous faire réveiller par un bug bête.
- Les validations sont les règles de la communauté

5. Raffinement: concepts avancés pour les ingénieurs

Ingénieur Logiciel 2.0 ✌️

- 5.0. Modèles de conception: Construire des bases SOLID
- 5.1. Automatisation des tâches Ne vous répétez pas
- 5.2. Crochets de pré-commit Votre CI/CD local
- 5.3. Flux de travail CI/CD Votre CI/CD partagé
- 5.4. Conteneurs logiciels Rendez-le reproductible
- 5.5. Expériences AI/ML: Suivi MLflow
- 5.6. Registres de modèles MLflow Model Registry



Automatisation
<https://xkcd.com/1319/>



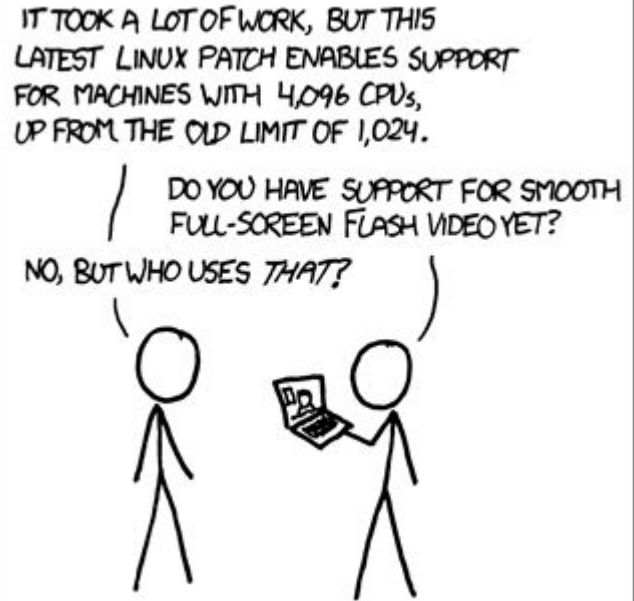
5. Affinement - Points clés

- Automatisation, automatisation, automatisation
- Les modèles de conception sont des solutions
prêtes à l'emploi
- Les conteneurs sont votre outil de prédilection
pour la reproductibilité.

6. Partage Construire avec les autres pour les autres

Rejoignez-nous maintenant et partagez le l

- 6.0. Dépôt Référez-vous à votre projet
- 6.1. Licence: Expliquez comment les gens peuvent l
- 6.2. Lisez-moi: Expliquez ce que les gens peuvent e
- 6.3. Versions Dites aux gens ce qui a changé.
- 6.4. Modèles: Apprendre aux gens comment constr propre
- 6.5. Stations de travail Aider les gens à exécuter vot
- 6.6. Contributions Dites aux gens comment contribuer



Fonctionnalités prises en charge
<https://xkcd.com/619/>

Besoin d'un coup de main ?

Virtuel ou réel, nous avons ce qu'il vous faut !

- Assistant chatbot de codage MLOps

Gemini avec l'intégralité du cours comme contexte, capable de répondre sur toute partie spécifique du cours.

Expliquer le choix des questions-réponses

- Mentorat





Place à la démo

Contributions et Suggestions

Le cours est Open Source sur Github

N'hésitez pas à le développer (mentionnez-nous juste ;))

Si vous avez des commentaires, des corrections ou de nouvelles idées de contenu, contactez-nous !

J'espère que ce cours aidera les apprenants à comprendre la
gestion des expérimentations et des modèles à travers
MLflow



© DALL.E

Merci de votre attention !

<https://mlops-coding-course.fmind.dev>

Rejoignez-nous sur le [Slack de la](#)
[Communauté MLOps](#)