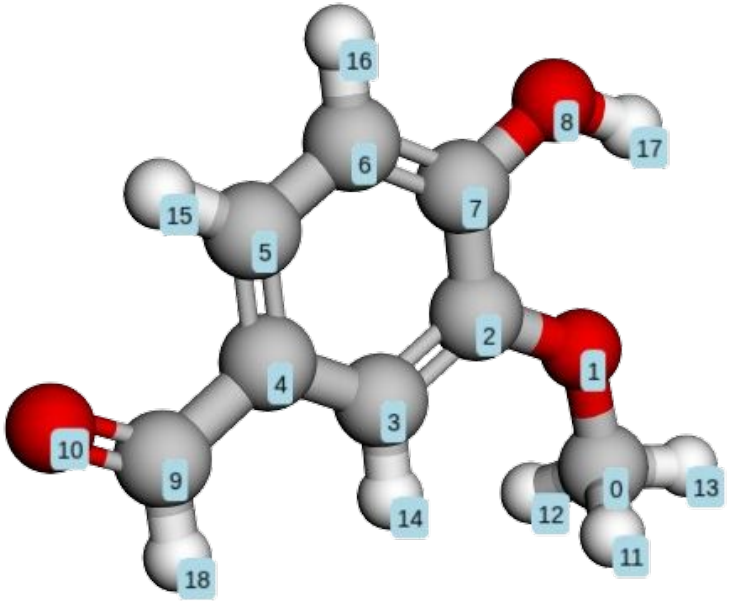
PredictionRMN

Stage M1 - LERIA
Université d'Angers

Stagiaire : Yamis MANFALOTI
Encadrant : M. Da Mota
Collaborateur : M. Cauchy

1. Introduction
2. Analyse de l’existant
3. Conception
4. Développement
5. Modèles de prédiction
6. Conclusion

Contexte Scientifique



COC1=CC(=CC=C1O)C=O

Illustration de la vanilline et de son SMILES

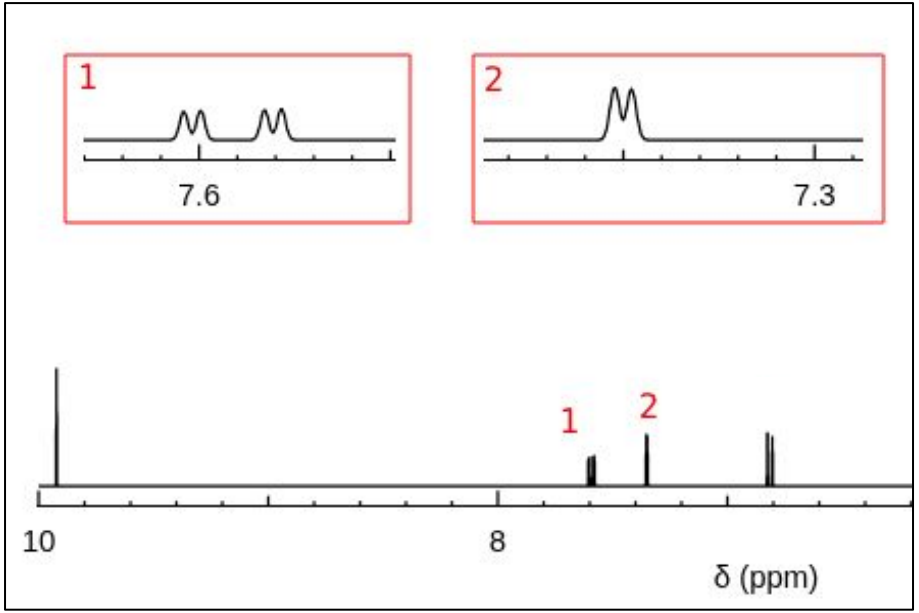
🧬 SMILES

- Format **textuel** pour décrire une molécule
- **Lisible** par l'humain et les logiciels
- Encodage sous forme de **chaîne de caractères**

Contexte Scientifique

📈 **Résonance Magnétique Nucléaire (RMN)**

- **Analyser la structure** d'une molécule
- **Deux atomes** : hydrogène et carbone
- Spectre :
  ppm et **intensité** des signaux
- Pics reflètent l'**environnement électronique** des atomes



Objectif

🧪 Objectif principal

- Conception et développement d'une **application ergonomique**
- **Visualisation de spectres prédits** à partir d'une molécule

🎯 Besoins identifiés

- Dessiner une **molécule** et visualiser le **spectre** correspondant
- Lire les **liens entre atomes et signaux**
- **Prédire le spectre** d'une molécule

Contraintes

🖥️ Contraintes ergonomiques

- Interface **claire**, **fluide** et **intuitive**
- **Facile à prendre en main** (utilisateurs non-techniques)

⚙️Contraintes techniques

- **Open source** (usage universitaire)
- **Compatible multi-plateformes** (Windows / Linux / Mac)

1. Introduction
2. Analyse de l’existant
3. Conception
4. Développement
5. Modèles de prédiction
6. Conclusion

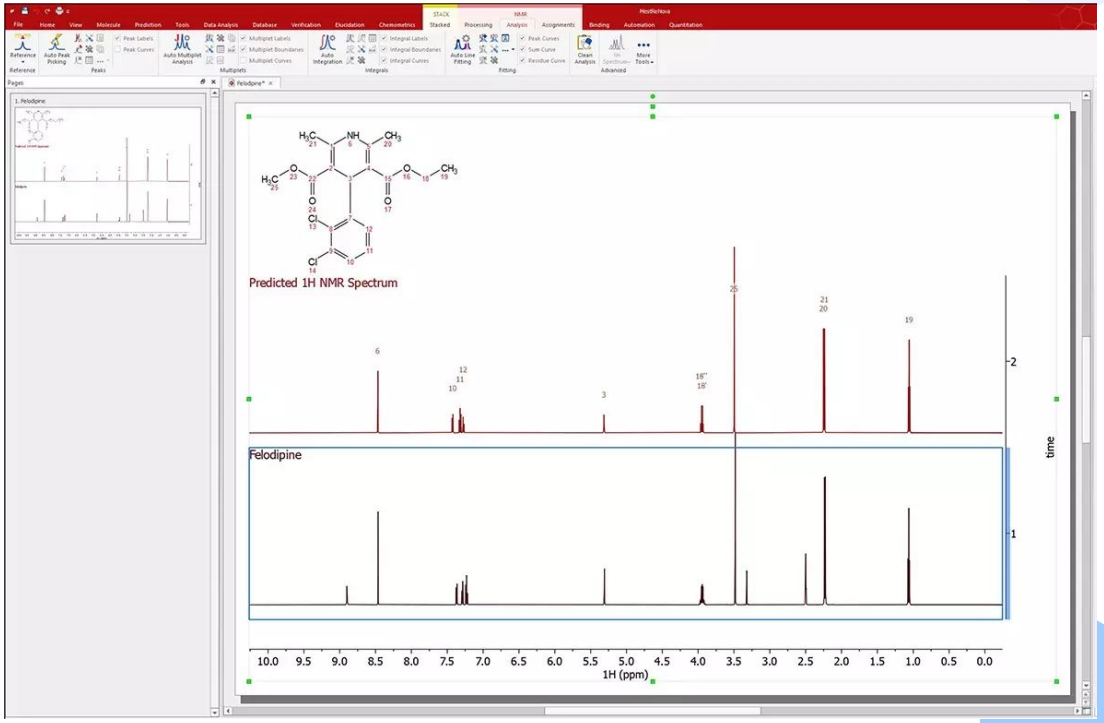
Solution Similaire

🔍 **Solutions étudiées :**

- NMRium (Zakodium)
- NMRPredict (Mestrelab)

🧪 **NMRPredict :**

- Référence **très utilisée par les chimistes**
- **Fonctionnalités** : dessin moléculaire, prédiction RMN, analyse de spectres
- **Logiciel propriétaire**



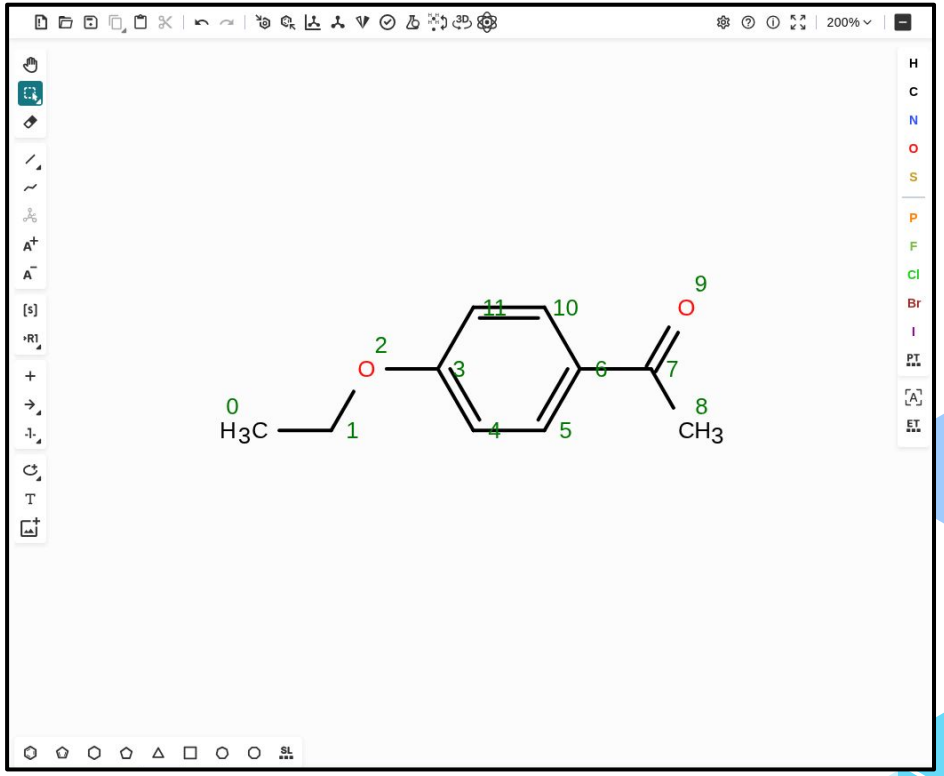
Composants Existants

🧩 **Composants étudiés :**

- 🧬 Un **éditeur moléculaire**
- 📊 Un **visualiseur de spectres** RMN
- 🖥️ Un **framework d'interface utilisateur**

✍ **Ketcher** (Epam) **:**

- **open source** (Licence Apache)
- **Moderne, complet et maintenu**
- **Permet le dessin de molécules et l'export en format SMILES**



1. Introduction
2. Analyse de l’existant
3. Conception
4. Développement
5. Modèles de prédiction
6. Conclusion

Méthodologie

🔄 Approche itérative

- **Échanges fréquents** avec le chimiste
- Flexibilité pour **s'adapter aux retours**
- Conception → prototype → test → retours

📚 Documentation technique

- Diagrammes, spécifications techniques, décisions de conception
- Modélisation **C4** et **UML**
- **Traçabilité** et **support** pour futurs développeurs

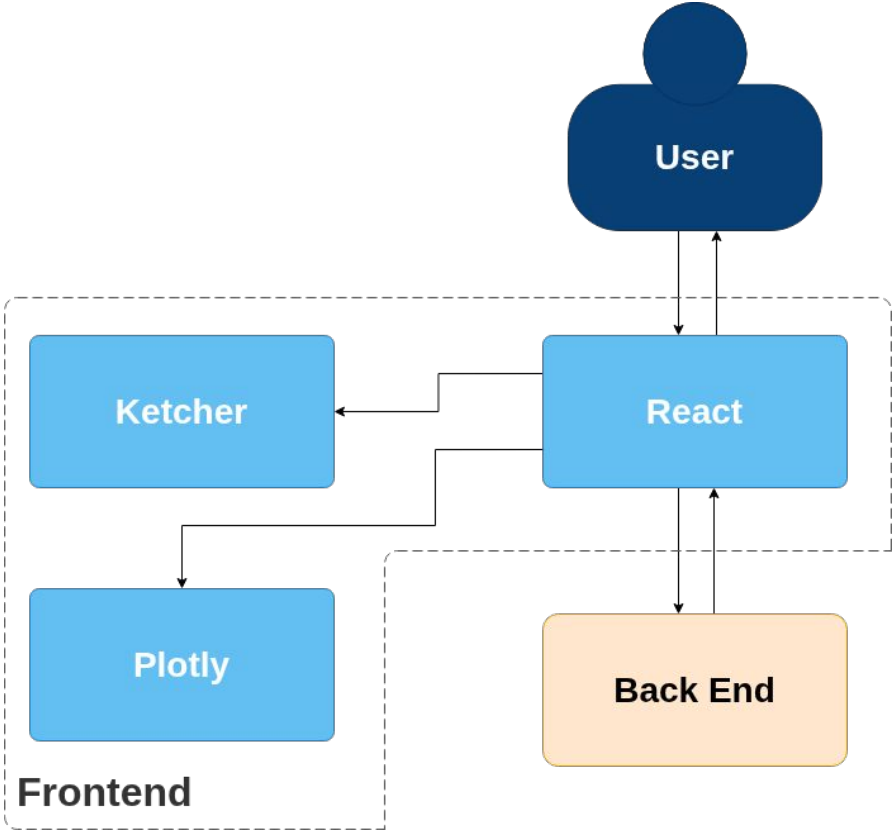
Architecture Frontend

🏗️ **Architecture Frontend :**

- Architecture **frontend / backend locale**
- **React** : interface moderne et modulaire

🧩 **Composants clés :**

- **Ketcher** : éditeur moléculaire
- **Plotly** : visualisation interactive des spectres



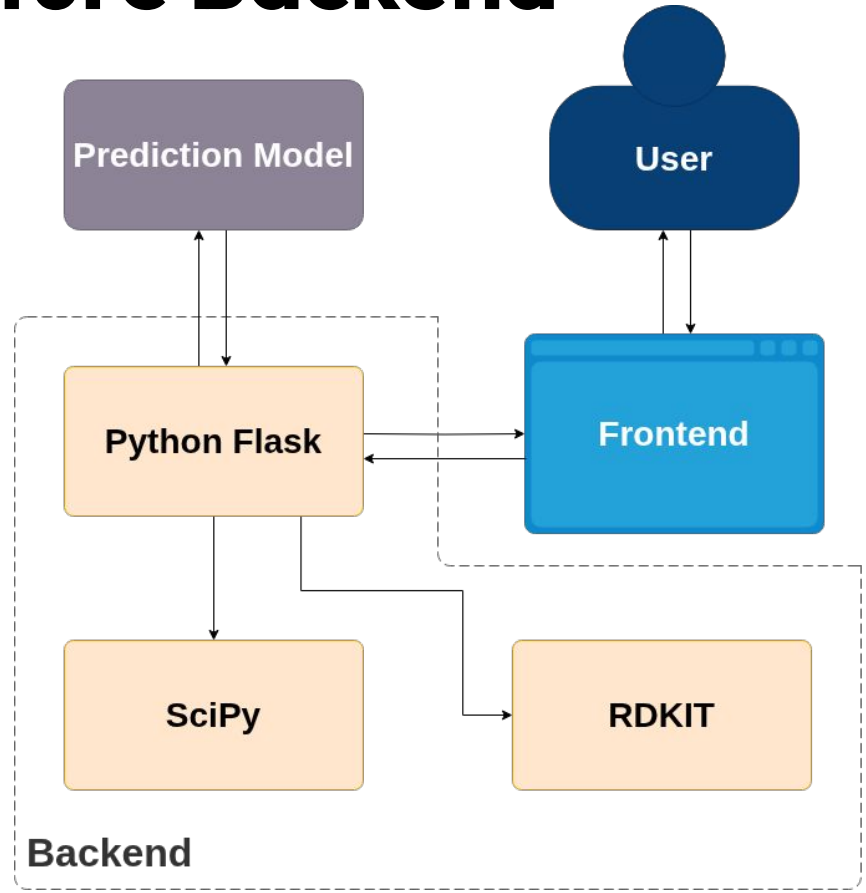
Architecture Backend

🏗️ Architecture Backend

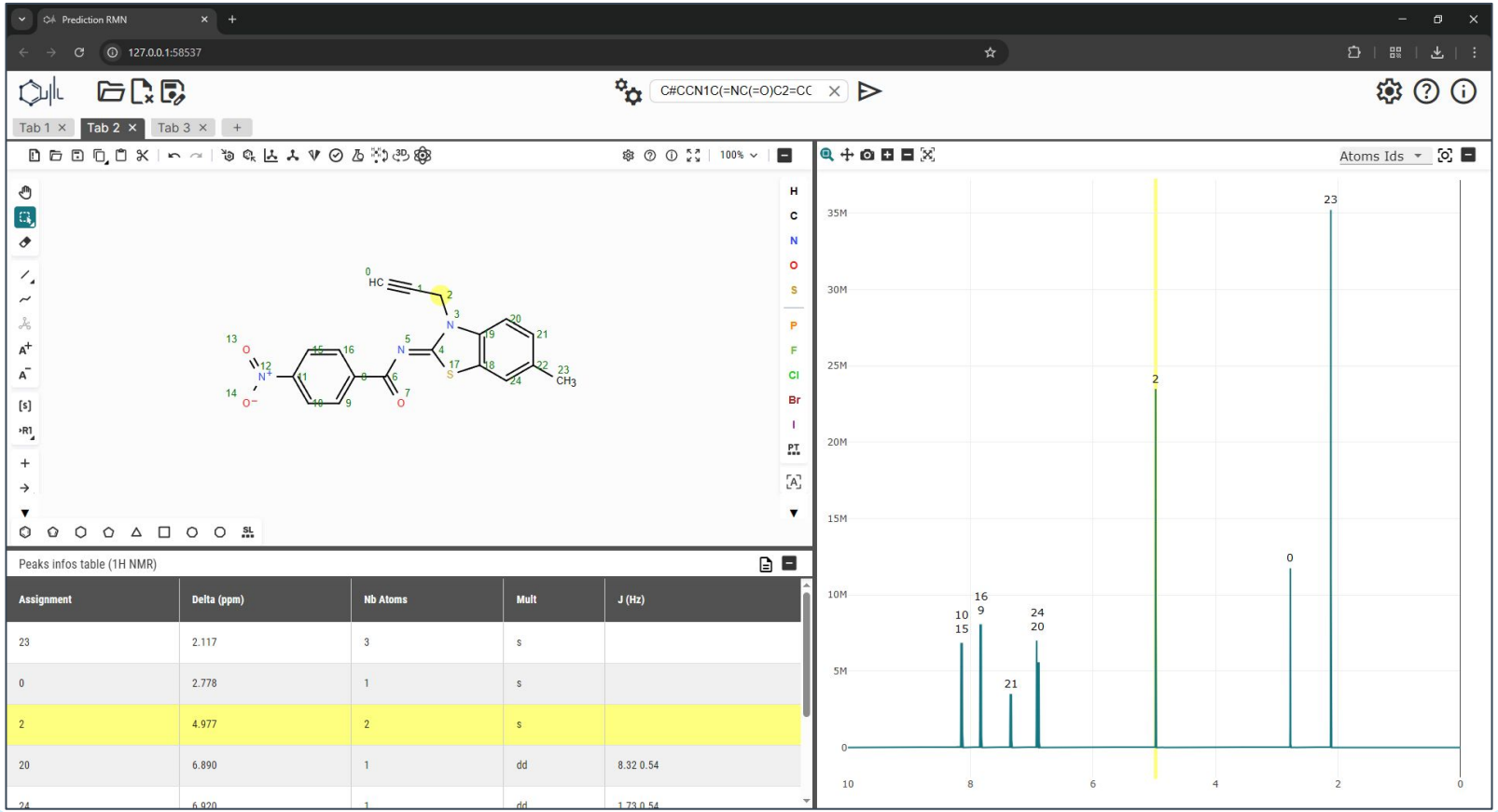
- **Python** : incontournable pour accéder aux outils chimiques
- **Flask** sert le **Frontend** et l'**API interne**
- **Gère la communication avec le modèle de prédiction**

🧪 Outils scientifiques intégrés :

- **RDKit** : traitement des molécules
- **SciPy** : traitement des spectres RMN



Interface


Prediction RMN
127.0.0.1:58537
C#CCN1C(=NC(=O)C2=CC
Tab 1
Tab 2
Tab 3
100%
Atoms Ids
Peaks infos table (1H NMR)
Assignment | Delta (ppm) | Nb Atoms | Mult | J (Hz)
23 | 2.117 | 3 | s
0 | 2.778 | 1 | s
2 | 4.977 | 2 | s
20 | 6.890 | 1 | dd | 8.32 0.54
24 | 6.920 | 1 | dd | 1.73 0.54

1. Introduction
2. Analyse de l’existant
3. Conception
4. Développement
5. Modèles de prédiction
6. Conclusion

Mise en place

🗃️ Dépôts github :

- Un dépôt pour l'**interface frontend** et la version **custom de Ketcher**
- Un dépôt pour le **backend** et l'**application finale**
- Un dépôt pour la partie **modèle de prédiction**

🛠️ Outils mis en place :

- **Prettier & ESLint :** formatage & détection erreurs style
- **Jest :** tests unitaires
- **Hooks Git (pre-commit & pre-push) :** vérifications automatiques (format, tests, typage)

Evolution des fonctionnalités

Itération 1 (10 jours)	Itération 2 (7 jours)	Itération 3 (7 jours)
Interface principale Affichage de Ketcher Affichage du spectre Simulation de prédiction	Persistance des paramètres Gestion des modèles de prédiction Système d'onglets	Exports du projet Sélection d'annotation du spectre Détection des pics et multiplicités du spectre

Déploiement et documentation

🖥️ **Déploiement :**

- **Compilation du frontend** React (fichiers statiques HTML, CSS, JS)
- **Tests** en conditions réelles
- **Packaging avec PyInstaller** (exécutables Linux, Windows, etc)

📚 **Documentation :**

- Mise à jour **documentation technique**
- Rédaction **guide utilisateur**
- Facilite la **prise en main** et la **maintenance**

1. Introduction
2. Analyse de l’existant
3. Conception
4. Développement
5. Modèles de prédiction
6. Conclusion

Dataset

❌ Absence de dataset :

- Aucun dataset public adapté
- Nécessité de créer notre propre dataset

🛠️ Méthode de collecte :

- Utilise le simulateur de nmrdb.org
- Scraping automatisé via Selenium

📊 Composition du dataset :

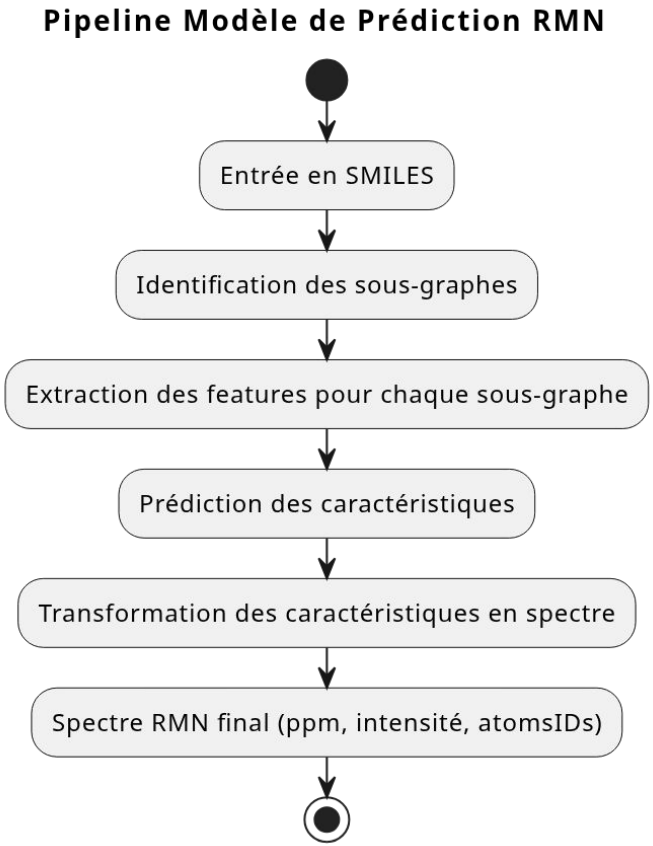
- 500 molécules extraites de PubChem
- Molécules organiques
- Deux datasets distincts (H et C)

🎯 Évolution :

- Augmentation à 1000 molécules
- Ajout de molécules non-organiques

Pipeline

- **Trop complexe** de prédire un spectre directement
- Le **découpage simplifie la tâche** en étapes claires
- Chaque étape **améliorable plus facilement**
- Assure un **spectre final plus précis et plus robuste**



Résultats

Critères	Modèle H (v3)	Modèle C (v2)
Erreur Moyenne (ppm)	**0.131** échelle [0,10]	**2.345** échelle [0,250]
RMSE (ppm)	0.317	5.557
Précision (multiplicité)	**85,4 %**	**-**
Points forts	**Spectre fidèle**, bonne précision globale	
Limites	**Erreurs occasionnelles** sur les multiplicités	**Données initiales limitées**

1. Introduction
2. Analyse de l’existant
3. Conception
4. Développement
5. Modèles de prédiction
6. Conclusion

Conclusion

🧠 **Ce que j'ai appris :**

- **Approche interdisciplinaire** (informatique / chimie)
- **Intégration de projets existants** (Ketcher, Plotly)
- **Bonnes pratiques de développement** : conception, conventions, CI, tests
- **Première vraie expérience** en modèle de prédiction (ML)

🔭 **Perspectives :**

- **Ajout et amélioration des fonctionnalités**
- **Amélioration des modèles** de prédiction

Merci pour votre écoute

CRÉDITS : Ce modèle de présentation a été créé par **Slidesgo**, et inclut des icônes de **Flaticon**, et des infographies et images de **Freepik**.

Conception UI / UX

🧾 Charte graphique

- Interface **centrée sur la lisibilité et l'ergonomie**
- Reprends la **charte graphique de Ketcher**
- Icônes de **Material Symbols** pour leur cohérence et leur lisibilité

#FFFFFF

#167782

#525252

#D9D9D9

Inter

Aa Bb Cc Dd Ee Ff... Zz