Manuel de déploiement

# Déploiement du LLM

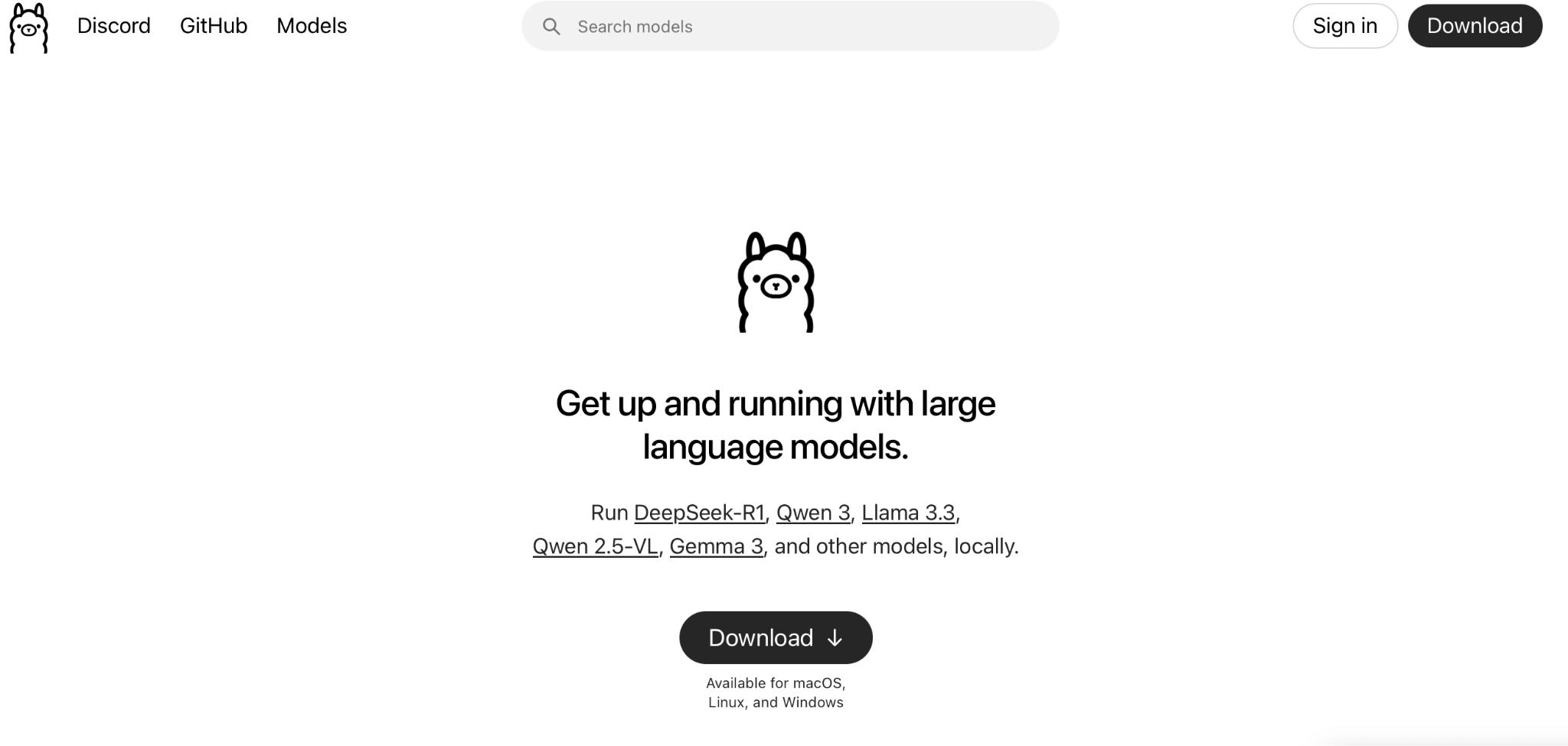
**Ollama** est une plateforme locale qui permet de télécharger, de gérer et d’exécuter des modèles d’IA (comme LLaMA, Mistral, Gemma, etc.) directement sur votre machine.

Elle offre une interface en ligne de commande simple, tout en prenant en charge l’exécution rapide de modèles d’inférence sans dépendre du cloud.

Ollama est particulièrement utile pour les systèmes de génération augmentée par récupération (RAG), ou tout usage nécessitant un contrôle total sur l’environnement d’exécution.

### Téléchargement et installation Ollama

Accédez au site d’Ollama et téléchargez la version adaptée à votre système d’exploitation.

  
  
*Fig 1. Page d'accueil Ollama*

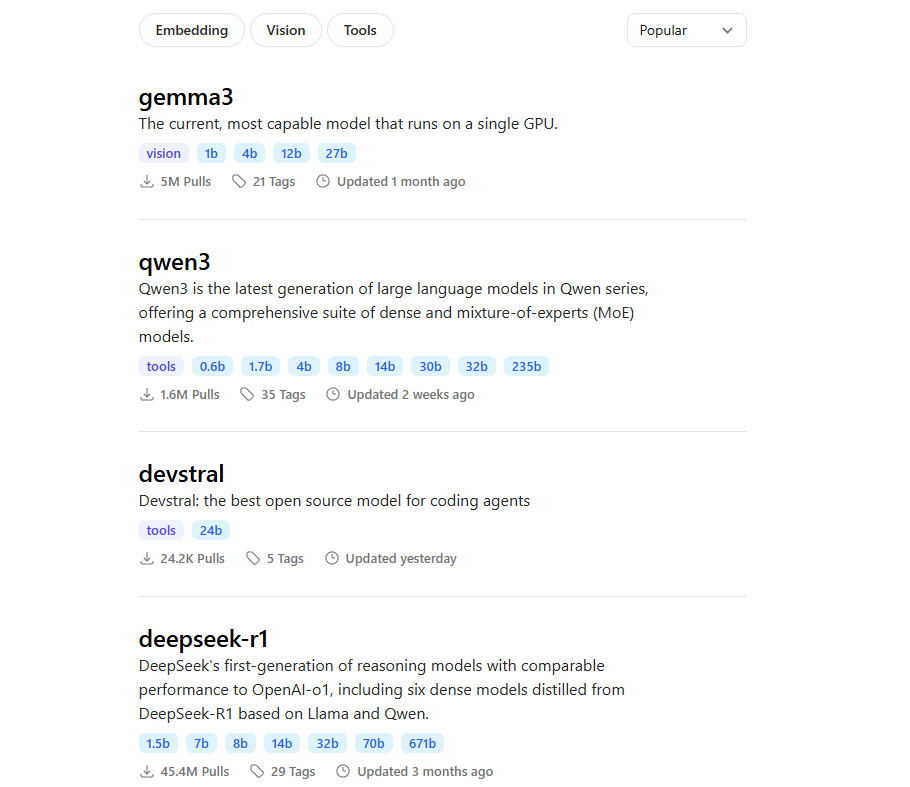
Le chemin d’installation étant prédéfini, il suffit simplement de cliquer sur **« Install »** pour lancer l’installation.  
  
Une fois l’installation terminée, ouvrez la ligne de commande à l’aide de **Windows + R**, puis entrez la commande suivante :

**ollama --version**

Cela permet de vérifier que Ollama a bien été installé correctement.

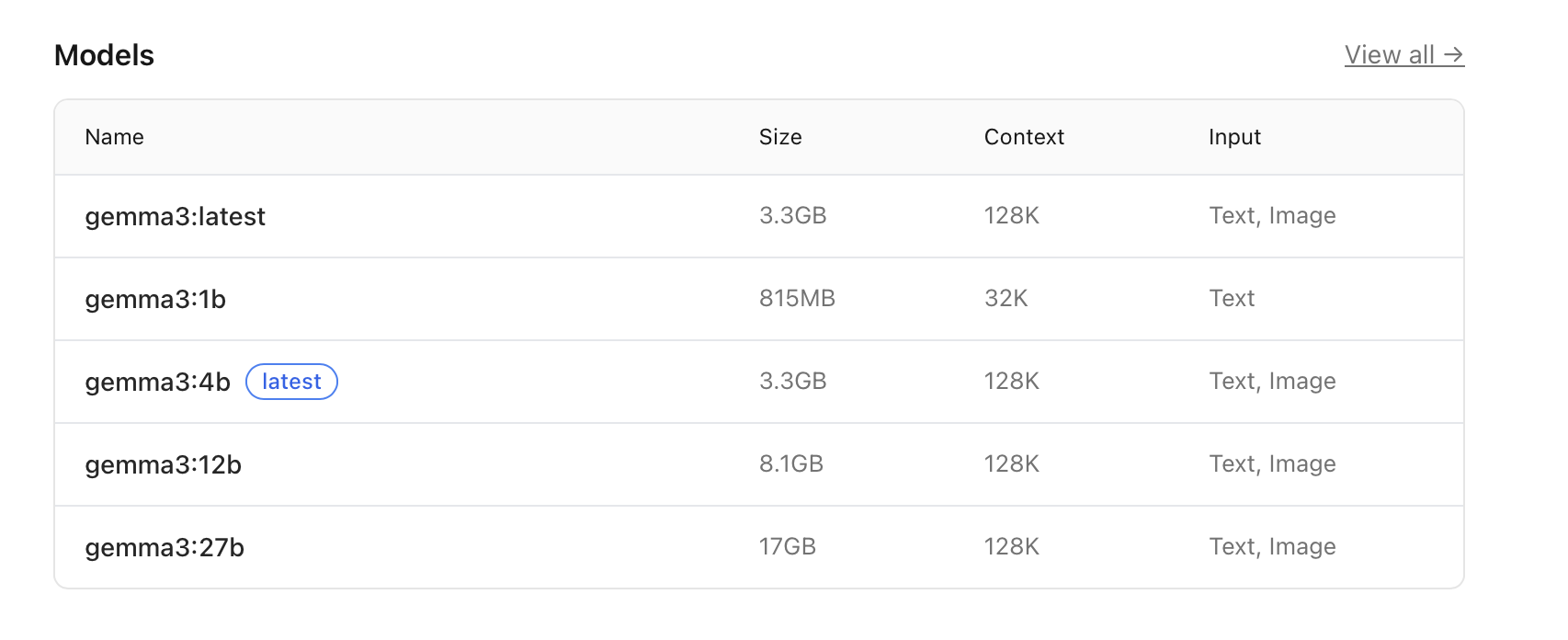
### Choix du LLM

Sur la page d’accueil d’Ollama, en cliquant sur **« Models »**, on peut consulter tous les modèles gratuits et disponibles. Sur la page dédiée à chaque modèle, on trouve toutes les informations correspondantes.

  
*Fig 2. Page des Models*

En fonction de vos ressources et de vos besoins, choisissez prudemment le **type** et la **taille** du modèle.Cela peut avoir un impact sur le temps de réponse ainsi que sur la qualité des réponses.

**Pour mettre en place ce projet, il est nécessaire de disposer d’au moins un modèle d’inférence et un modèle d’embedding.**  
Par exemple, sur la page du modèle Gemma 3, vous pouvez consulter la section **« readme »** pour obtenir des informations détaillées sur le modèle.  
Chaque modèle dispose d’une fiche technique listant ses principales caractéristiques, comme le nombre de paramètres, la taille de la fenêtre de contexte (*context window*), et les ressources requises.  
  
En général, plus un modèle possède de paramètres, plus sa fenêtre de contexte est grande. Cela permet de traiter des entrées plus longues et d’améliorer la pertinence des réponses. Ces modèles tendent également à offrir une meilleure qualité de génération, bien que cela puisse dépendre du domaine d’application.  
En contrepartie, les modèles plus volumineux nécessitent davantage d’espace disque pour être stockés, ainsi qu’une quantité significative de mémoire vive (RAM) au moment de l’exécution.

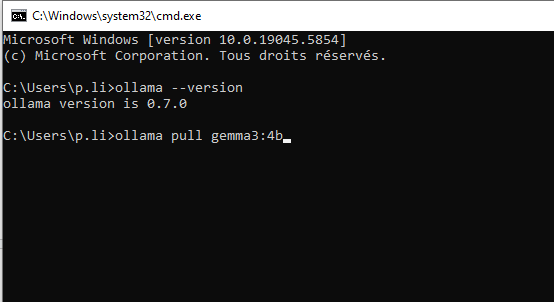
  
*Fig 3. Taille d’un modèle*

Si vous n’avez pas de carte graphique NVIDIA, vous ne pourrez pas bénéficier de l’accélération matérielle via CUDA. L’inférence sera alors effectuée uniquement par le processeur (CPU), ce qui peut entraîner une charge élevée et ralentir d’autres tâches sur votre machine.

### Installation du LLM

Une fois le LLM défini, ouvrez la ligne de commande, puis entrez la commande suivante afin d’installer le modèle souhaité :

**ollama pull <nom complet du modèle>**

  
*Fig 4. Téléchargement d’un modèle*

Si la taille du modèle (*model size*) n’est pas spécifiée explicitement, la version par défaut indiquée sur la page du modèle sera utilisée.  
  
Si vous souhaitez utiliser votre propre modèle, il est également possible de l’importer dans Ollama.Pour plus de détails, consultez la documentation officielle : <https://github.com/ollama/ollama>

1. Désinstallation d’un modèle  
   Si vous souhaitez désinstaller un modèle, entrez la commande suivante :

**ollama rm <nom du modèle>**

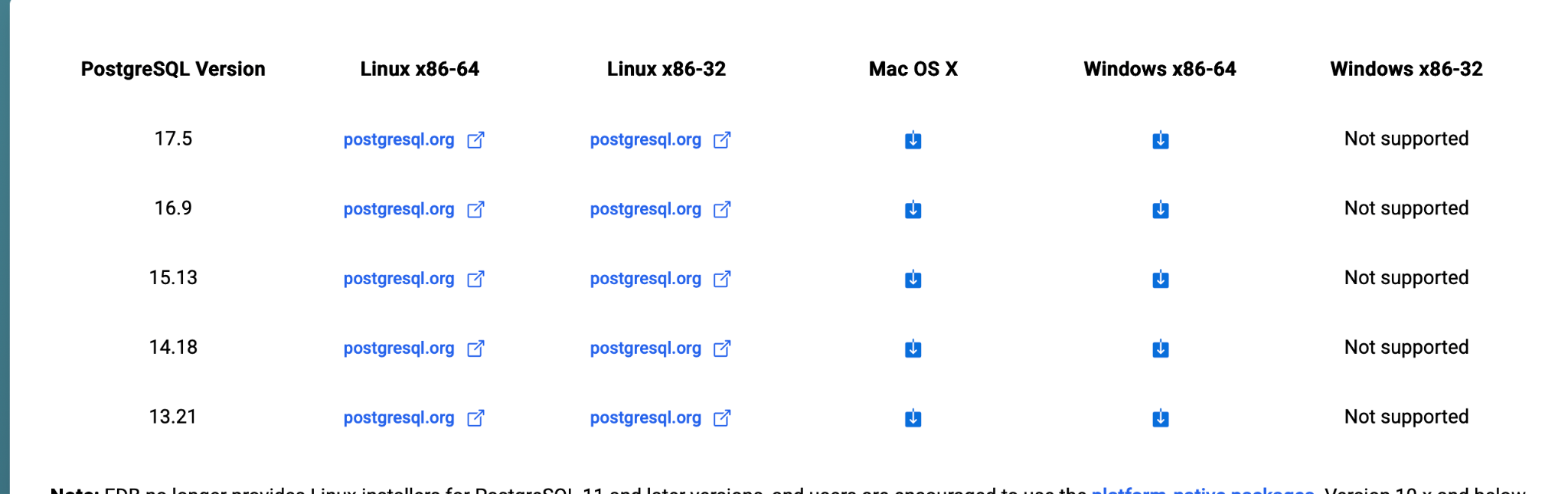
Pour plus d’informations, utilisez la commande :

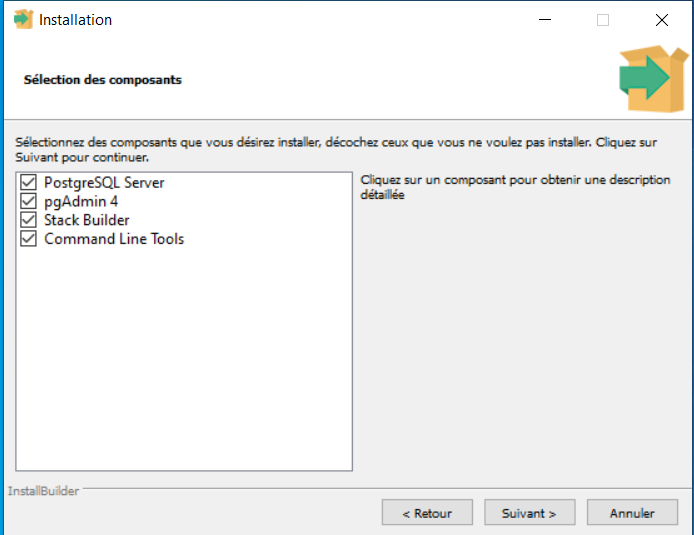
**ollama -h**

# Déploiement de la base de données

Dans le cadre de ce projet, PostgreSQL est utilisé pour **stocker les documents indexés** ainsi que leurs **représentations vectorielles** générées à partir de modèles d’embedding.  
  
**pgvector** est une extension de PostgreSQL qui permet de stocker et de rechercher des vecteurs. Elle est essentielle pour effectuer des recherches sémantiques basées sur des similarités vectorielles.

### Installation de PostgreSQL

* Accédez au site [Download PostgreSQL](https://www.enterprisedb.com/downloads/postgres-postgresql-downloads) téléchargez la version supérieur à 15  
    
    
  *Fig 5. Téléchargement de PostgreSQL*
* Après avoir spécifié le chemin d’installation, on doit sélectionner les composants qu'on souhaite installer. L’installation de tous les composants sont recommandés.Toutefois, si vous êtes déjà familier avec PostgreSQL, vous pouvez adapter l’installation selon vos besoins.

  
*Fig 6. Installation de PostgreSQL*

| Nom du composants | Utilité principale | Remarques |
| --- | --- | --- |
| PostgreSQL Server | Moteur de base de données | Indispensable |
| pgAdimin 4 | interface graphique pour la gestion de la base de données |  |
| Stack Builder | Outil d’installation d’extensions tierces et de composants supplémentaires | installe PostGIS, pgAgent ,etc. |
| Command Line Tools | Outils en ligne de command (psql, pg\_dump, etc. ) | Utilisés pour les scripts, l’automatisation |

*Tab 1. Les Composants de PostgreSQL*

* Dans les étapes suivants, vous devez compléter la configuration des paramètres de la base de données :   
    
  **Mot de passe du superutilisateur** ,le seul compte disponible lors du premier démarrage de la base, d’autres comptes pourront être créés ultérieurement.  
    
  **Port d’écoute,** utilisé pour permettre les connexions des clients à la base de données.

*A noter que les données de la base sont stockées dans un répertoire séparé, lequel pourrait être défini lors de l’installation. En cas de désinstallation ultérieure de PostgreSQL, ces données ne seront pas supprimées automatiquement, ainsi que les extensions non gérés par Stack Builder.*

### Configuration de PostgreSQL

PostgreSQL peut être utilisé de deux façons :

1. En **ligne de commande** avec l’outil `psql`, adapté aux utilisateurs techniques.

2. Via une interface graphique conviviale avec `**pgAdmin**`, recommandé pour les utilisateurs préférant une gestion visuelle.

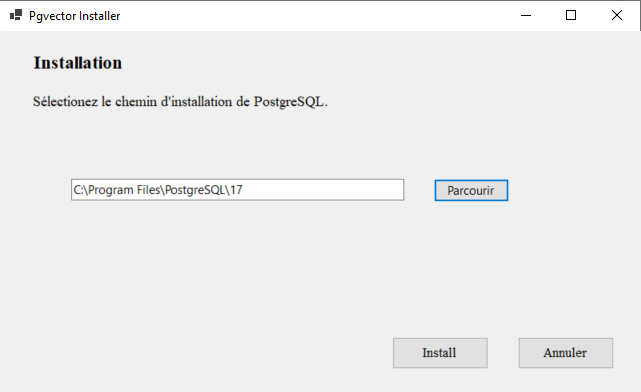
Pour les utilisateurs, `pgAdmin` offre une manière intuitive d’explorer la base. Les utilisateurs avancés préféreront probablement `psql` pour sa rapidité et son efficacité.

### Installation de pgvector

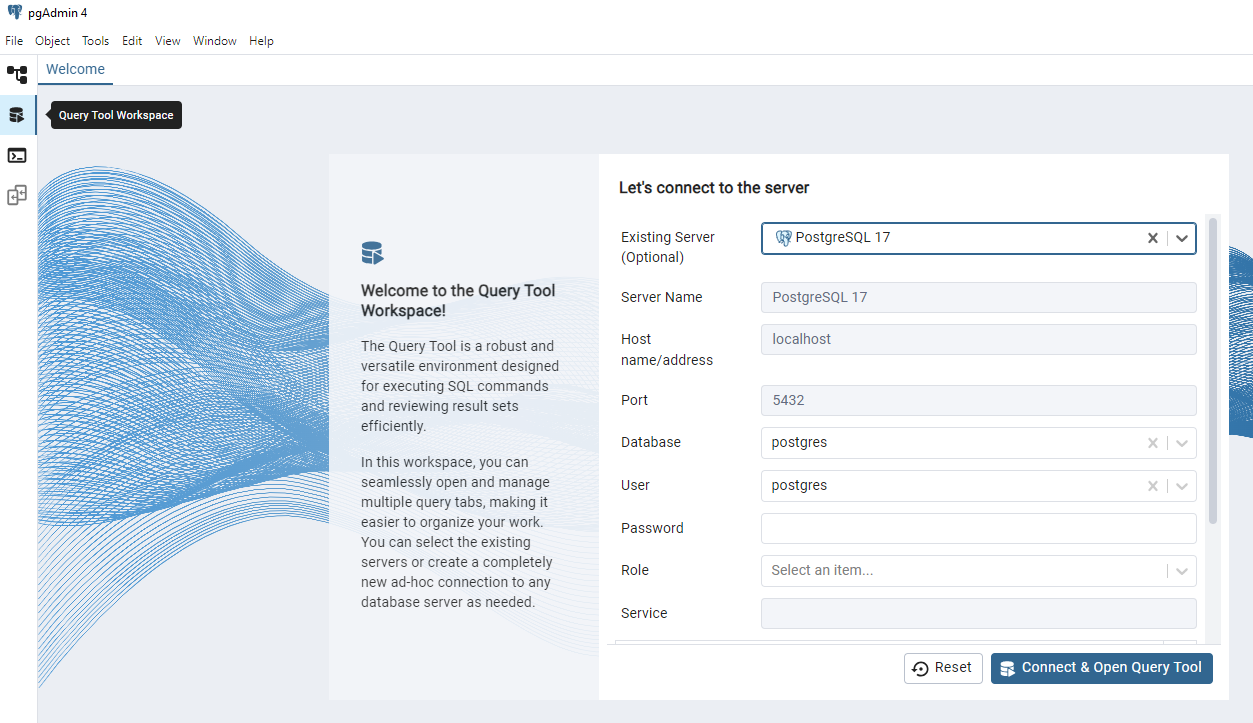
pgvector est une **extension PostgreSQL** qui permet de stocker, comparer et rechercher des **vecteurs** dans une base de données relationnelle.

1. Exécutez l’application Pgvector Installer et sélectionnez **le répertoire racine** de votre installation PostgreSQL.

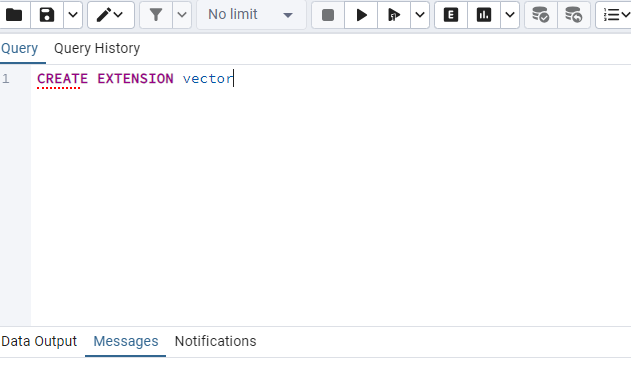
**Attention** : Pour installer l’extension, vous devez exécuter le programme d’installation (Pgvector Installer) **avec les droits d’administrateur**. Sinon, l’application ne pourra pas copier les fichiers nécessaires dans le répertoire d’installation de PostgreSQL, ce qui entraînera l’échec de l’installation.

  
*Fig 7. Installation de pgvector*

1. Connectez-vous à votre base de données PostgreSQL à l’aide de votre outil préféré (par exemple, pgAdmin ou psql), puis exécutez cette commande une fois dans chaque base de données où vous souhaitez activer l’extension vector :  
    **CREATE EXTENSION vector**



*Fig 8. Connexion de PostgreSQL via PgAdmin 4*

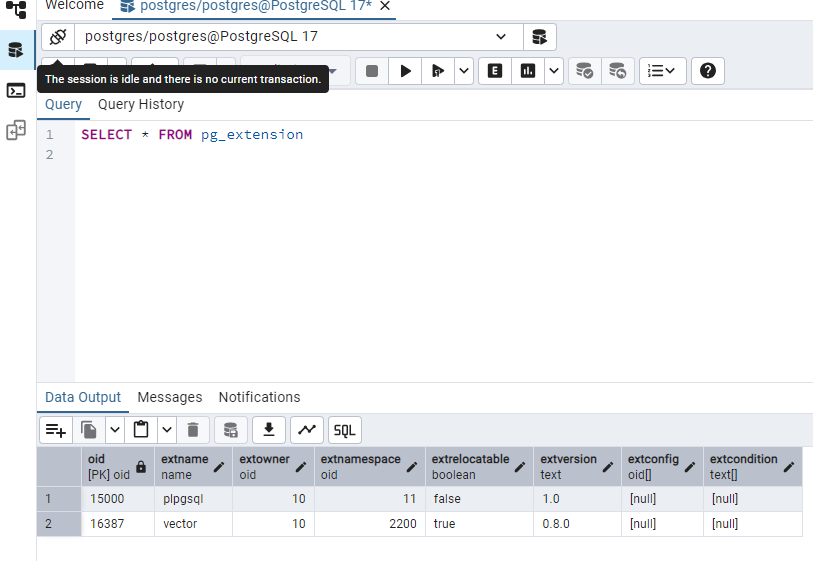


*Fig 9. Activation d’extension*

3. Si, après avoir exécuté cette commande, le nom de l’extension “**vector**” apparaît dans la liste des extensions affichée par la requête suivante :

**SELECT \* FROM pg\_extension;**

alors l’extension a été installée avec succès.



*Fig 10. Vérification d’installation*

# Téléchargement du code

Une fois tous les composants prêts, il suffit de télécharger le code pour pouvoir l’utiliser sur <https://github.com/Ar1shadow/RagAI_v2>.