

# Chatbots en python

## Partie 2: diagrammes d'états

Nathanaël Khodl

# Doc de la librairie

— — —

- Testés par les développeurs
- Mis à jour avec les évolutions
- Offre +/- exhaustive des possibilités

[docs.python-telegram-bot.org](https://docs.python-telegram-bot.org)



Eldinnie and jsmnbom Persistence (#1017) ...

..

|                                              |                                    |
|----------------------------------------------|------------------------------------|
| <a href="#">LICENSE.txt</a>                  | dedicate examples to the public    |
| <a href="#">README.md</a>                    | Persistence (#1017)                |
| <a href="#">conversationbot.png</a>          | [ci skip] add state diagrams for c |
| <a href="#">conversationbot.py</a>           | Revert "Context based callbacks    |
| <a href="#">conversationbot2.png</a>         | [ci skip] add state diagrams for c |
| <a href="#">conversationbot2.py</a>          | fixes comment on examples/con      |
| <a href="#">echobot.py</a>                   | Update examples with consisten     |
| <a href="#">echobot2.py</a>                  | Revert "Context based callbacks    |
| <a href="#">inlinebot.py</a>                 | Revert "Context based callbacks    |
| <a href="#">inlinekeyboard.py</a>            | Revert "Context based callbacks    |
| <a href="#">passportbot.html</a>             | Prepare for 11.0.0 release         |
| <a href="#">passportbot.py</a>               | Bot api 4.1 (#1198)                |
| <a href="#">paymentbot.py</a>                | Revert "Context based callbacks    |
| <a href="#">persistentconversationbot.py</a> | Persistence (#1017)                |
| <a href="#">timerbot.py</a>                  | Revert "Context based callbacks    |

# Analyse d'une conversation banale

— — —

\*UN CLIENT ENTRE\*

- **Bonjour, c'est pour boire un verre ou pour manger?**

[ATTENTE DE LA REPONSE]

- **“Manger”**
- **D'accord! Je vais vous chercher des menus et je reviens vous installer.**

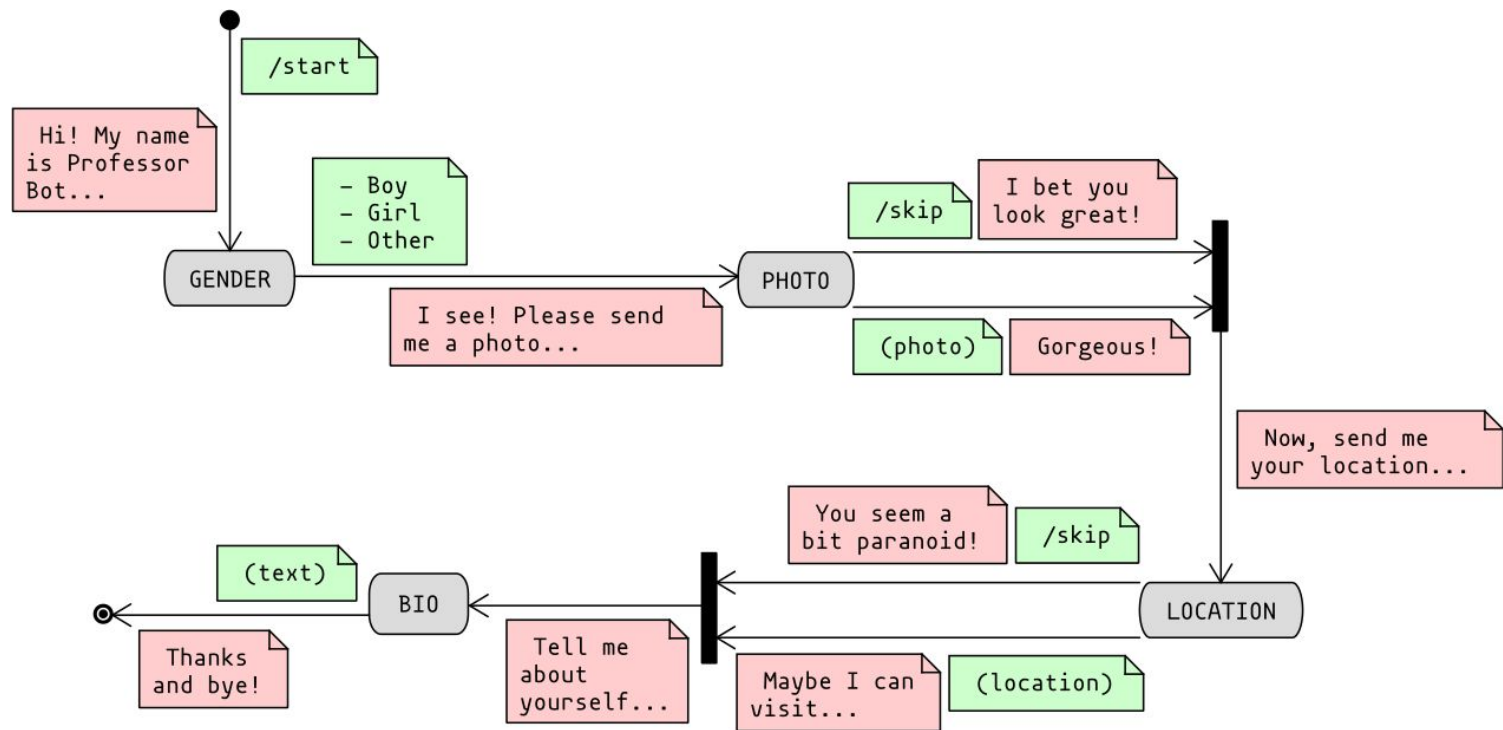
\*UN CLIENT ENTRE\*

- **Bonjour, c'est pour boire un verre ou pour manger?**

[ATTENTE DE LA REPONSE]

- **“Boire”**
- **Très bien! Nous avons des jus de fruit et des sodas. Que voulez-vous ?**

# Exemple de diagramme d'états

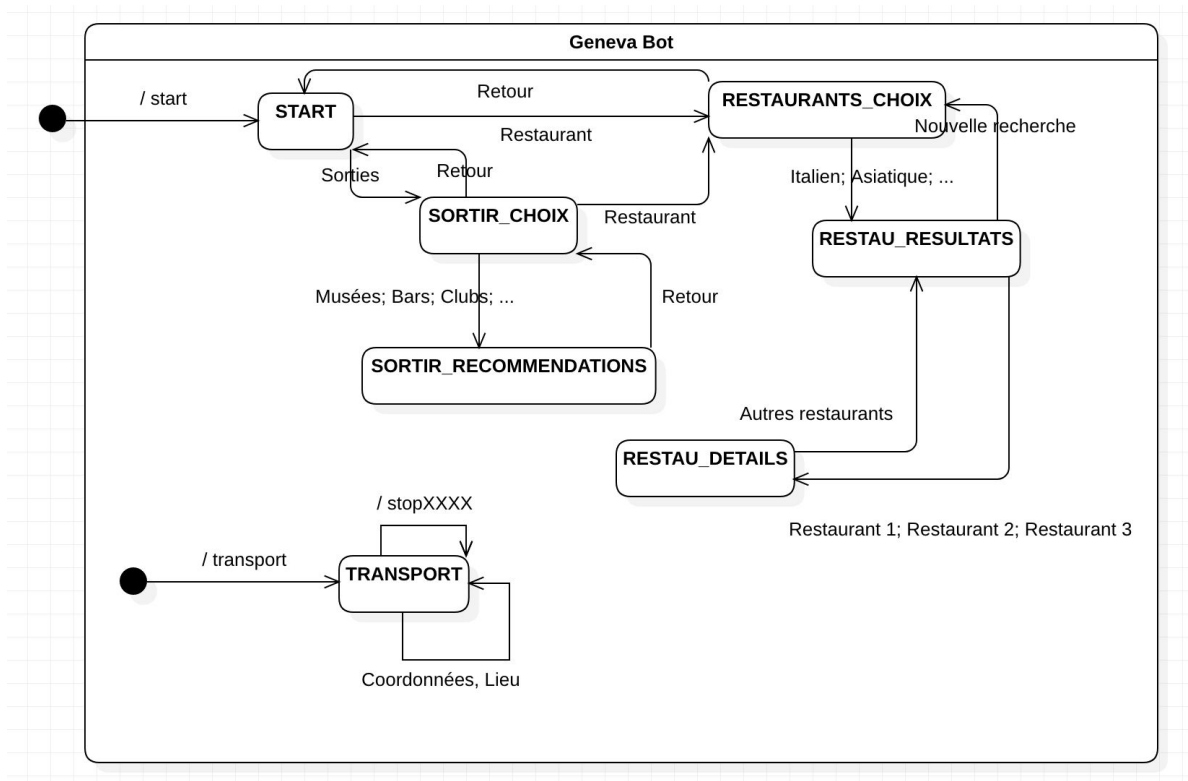


# conversationbot.py

— — —

1. Conversation handler VS updater.dispatcher
2. “Return WELCOME”
3. RegexHandler
4. ReplyKeyboardMarkup, un tableau de tableaux

# Exercice noté



# Consignes et notation

---

## **Implémentation de la partie du haut**

35 points (7 par état, START compris)

→ Tout doit être guidé (avec clavier, pas de texte libre)

## **Implémentation de la partie des transports**

10 points

→ Réimplémenter le bot que nous avons fait précédemment

## **Bot agréable à utiliser**

15 points

→ Cf Notre document “*Bonnes pratiques chatbots*”.

# Données à afficher

— — —

- Sorties
  - Les catégories sont: Musées, Bars, Clubs
  - Chaque catégorie doit afficher un top 3, chaque résultat étant dans un message. Les top 3 sont différents et correspondent à chaque catégorie.
- Restaurants
  - Choisissez 5 types de cuisine sur local.ch
  - Peu importe le type de cuisine choisi, affichez une liste de restaurants
  - Peu importe le restaurant choisi, affichez les détails d'un restaurant. Affichez le restaurant sur une carte.



# Pour rendre

---

Poussez votre script sur un repository git sur Gitlab ou Github, et donnez les accès à @khodl (avec les accès suffisants)

Le script doit pouvoir être lancé avec la commande suivante:  
→ **python chatbot\_etats.py TOKEN**

**Délai: 28 janvier à 23:59 CET**  
**Rendu: sur [gitlab/github](#)**