

Présentation de Tock



TOCK

Sommaire

- Introduction aux bots
- La technologie Tock
- Architecture générale d'un ChatBot
- Les composants applicatifs de Tock
- Architecture de Tock
- Les concepts de développement
- Démo
- A savoir

Introduction aux bots

I Introduction aux bots

- Programme informatique qui accomplit des tâches automatiques et répétitives.
- Agent intelligent
- Est virtuel et non physique comme des robots

I Introduction aux bots : Types

Un robot

- Analyse, automatisation de tâches

Exemple :

- Robots de fraude stream spotify
- Robots indexation chez Google
- Robots qui cliquent sur Youtube

I Introduction aux bots

- Un **chatbot** est un Bot programmé pour dialoguer avec les visiteurs
- Agent conversationnel
- Permet de s'exprimer dans un **langage naturel**
- Se met au service des interlocuteurs (**Pull**)
- Sollicite des interlocuteurs (**Push**)
- Application informatique avec **logique métier**

I Introduction aux bots

- Apportent des **besoins** en nouveaux métiers clés (PO conversationnel , UX)
- Permet de **traiter plus de demandes** que des humains dans un même laps de temps
- Analyser des demandes prioritaires

Introduction aux bots

De quand date le 1er Bot ?

Les années 1960

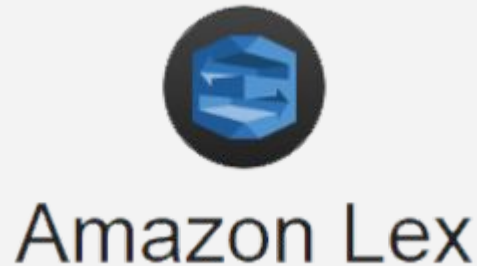
ELIZA

Simulation d'entretien avec un
psychothérapeute.



I Introduction aux bots : BotBuilders

Kaibee



La technologie Tock

I TOCK

- Plateforme complète pour construire des [chatbots](#)
- Open source (licence Apache 2)
- Utilise des libs Open source
- Créé par Oui sncf

I TOCK (Avantages)

- Ne dépend pas d'API tierces
- Maîtrise des modèles et des données.
- Multi-connecteurs (Facebook , Whatsapp etc)
- FAQ Ready
- Interface de gestion
- Multi-Language (API) ou langage récents (Kotlin)
- Testable

I TOCK (Limites)

- Documentation limitée
- Récent (première version en 2017)
- Communauté moins importante que d'autres bot builders

I TOCK (Connecteurs)



Messenger



Slack



Google
Assistant



Alexa

Kaibee

I TOCK (Connecteurs)

Kaibee



Messenger



Slack



Google
Assistant



Alexa



Teams



Apple BChat



Twitter



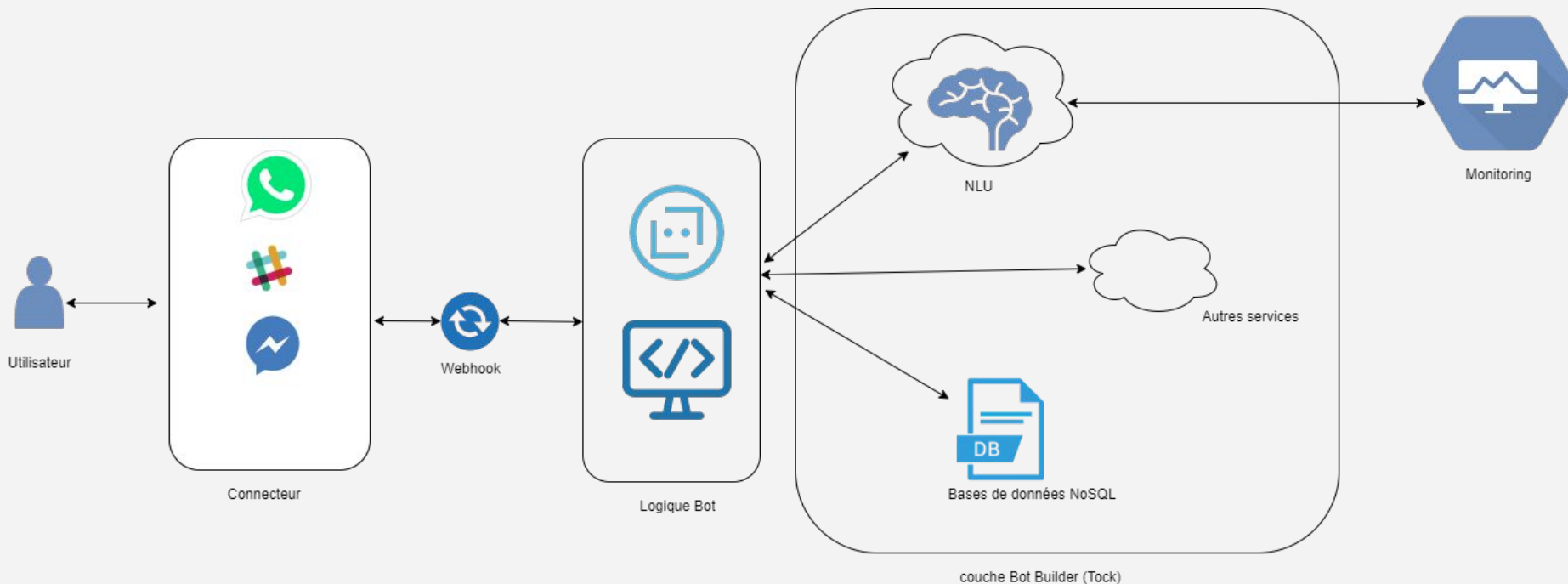
WhatsApp



Rocket Chat

Architecture générale d'un ChatBot

Architecture générale des chatbots

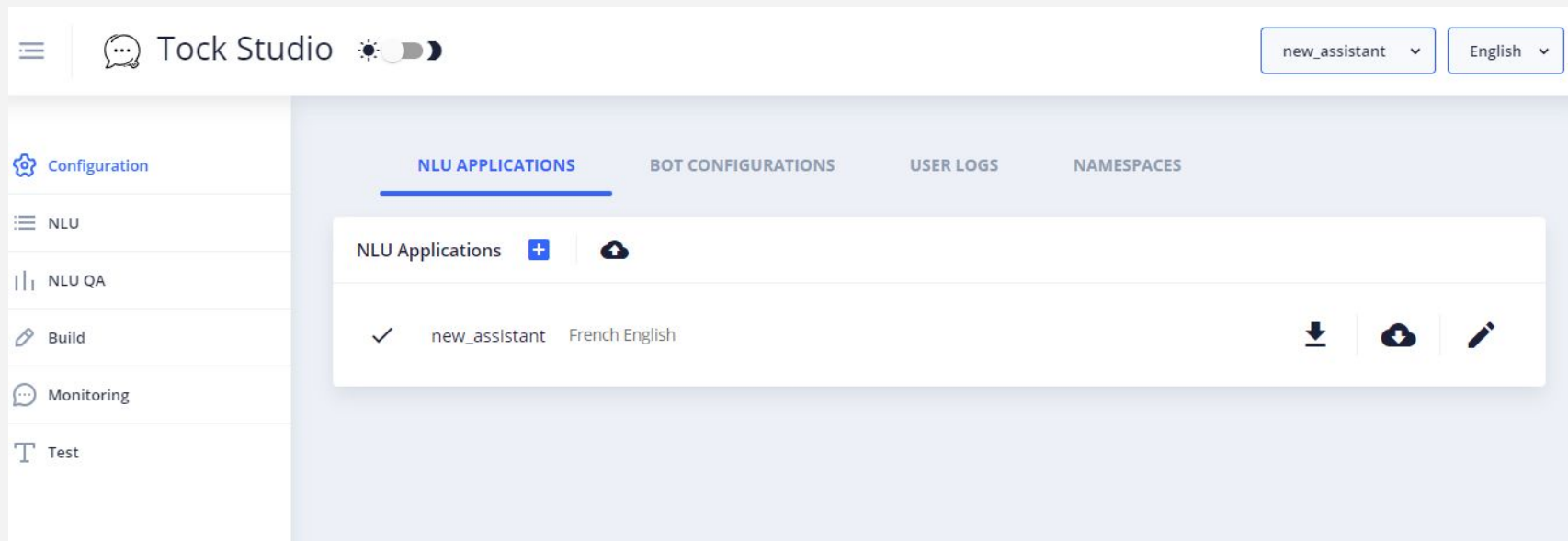


Les composants applicatifs de Tock

Les composants applicatifs de Tock

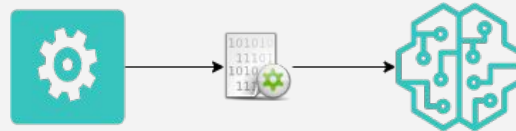


Tock Studio / Bot Admin



Les composants applicatifs de Tock

- Build Worker
Construit le modèle

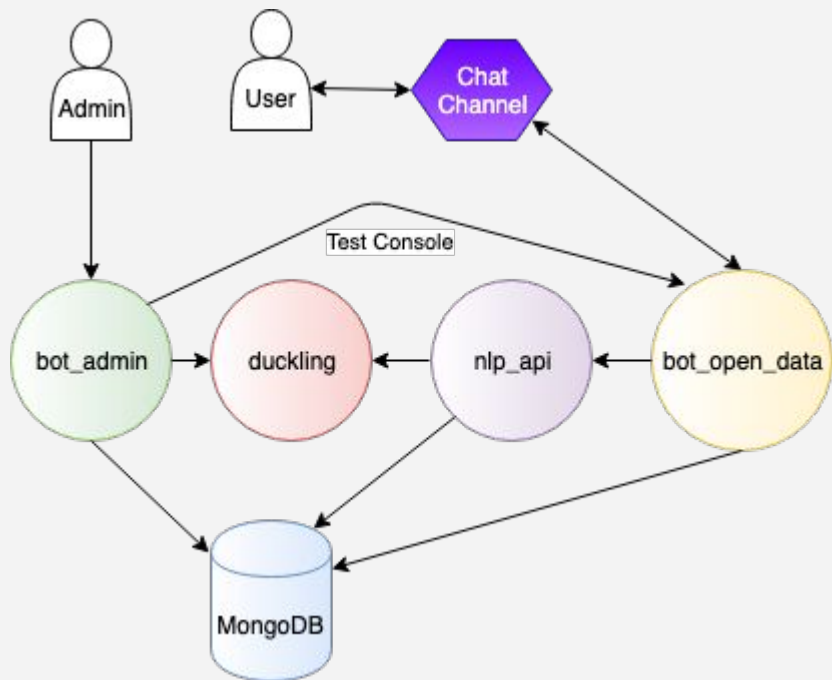


- NLP API
Analyse les phrases à partir du modèle
- Duckling
Analyse les types primitifs

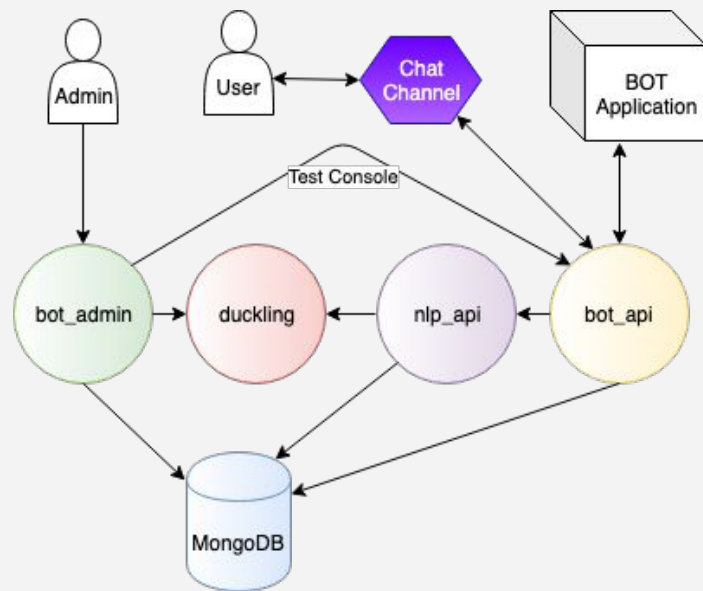
Architecture de Tock

Architecture de Tock

Le mode Bot Intégré (Kotlin)



Le mode Bot API



I Architecture de Tock : Autres

VERT.X

- Serveur de type "event driven" (réaction aux différents événements qui peuvent se produire)
- Tourne sur JVM
- Adapté pour gérer un bus d'évènement (Isolation du code / échange des informations entre entités)

I Architecture de Tock : Autres



- Serveur de type "event driven" (réaction aux différents événements qui peuvent se produire)
- Tourne sur JVM
- Adapté pour gérer un bus d'évènement (Isolation du code / échange des informations entre entités)

Les concepts de développement

Les concepts de développement

- **Namespace**
Pour isoler les groupes organisationnels dans tock (applications, objets)
- **Application**
Logique métier et entité pour la gestion d'une conversation avec des utilisateurs
- **Connecteur**
Canal conversationnel comme par exemple Messenger

Les concepts de développement

- Intention

Demande de l'utilisateur classifiée à partir de phrases utilisateurs

- Entité

Mot clé à signification particulière avec un rôle

Ex: Demain , aller , retour

- Story

regroupement fonctionnel qui permet de répondre aux questions sur un sujet bien délimité.

Démo

Développement de story avec Tock Admin



Configuration des applications

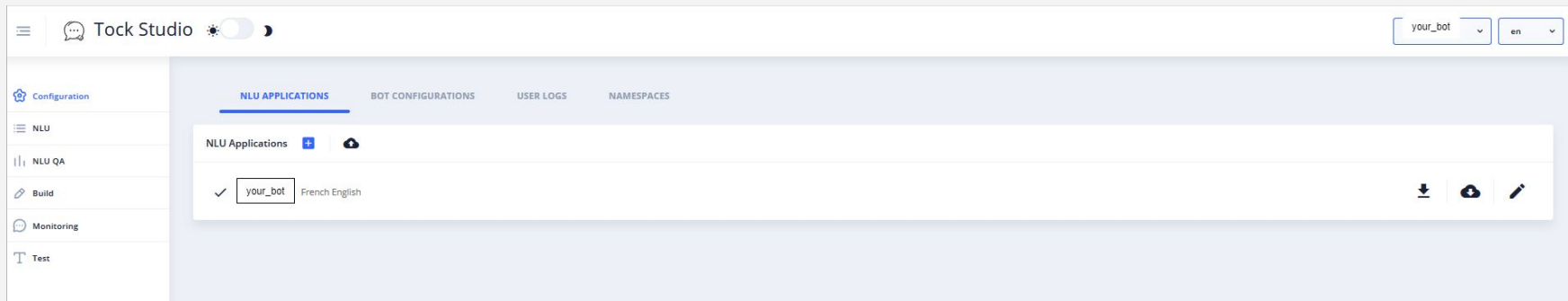
The screenshot displays the Tock Studio web interface. At the top, the header includes the "Tock Studio" logo, a theme toggle (light/dark), and dropdown menus for the selected bot ("your_bot") and language ("en"). The left sidebar contains a navigation menu with options: Configuration (highlighted), NLU, NLU QA, Build, Monitoring, and Test. The main content area is titled "NLU APPLICATIONS" and includes sub-tabs for "BOT CONFIGURATIONS", "USER LOGS", and "NAMESPACES". Below these, a table lists the configured NLU applications. One application, "your_bot", is shown with a checkmark, a plus icon, a cloud icon, and the language "French English". Action icons for download, upload, and edit are visible at the end of the row.

NLU Applications	
✓	your_bot French English

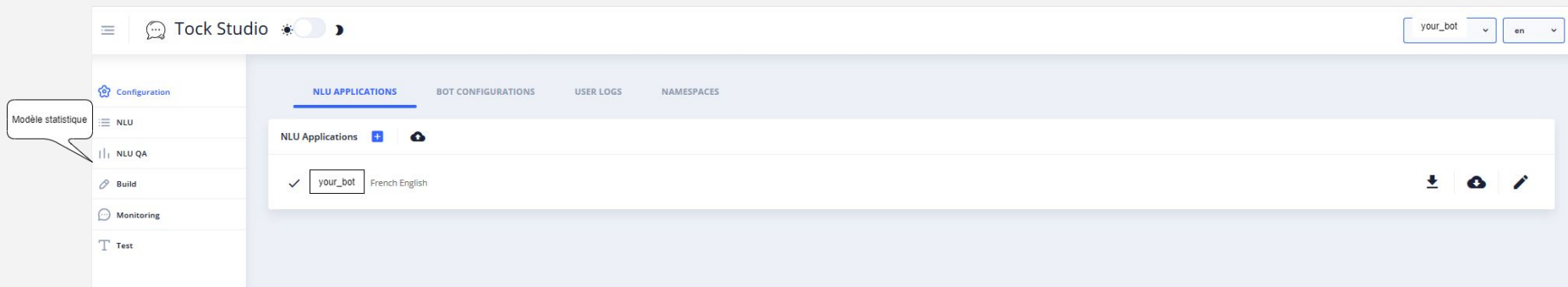
Développement de story avec Tock Admin



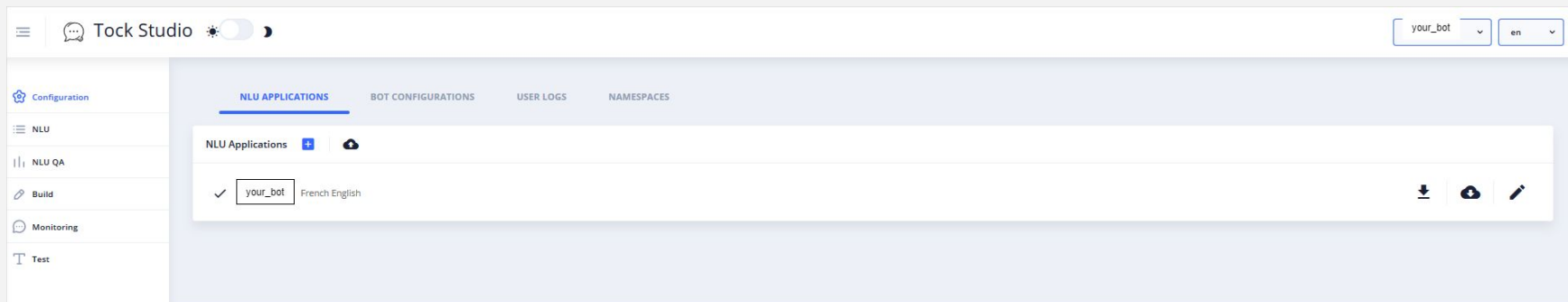
Gestion du modèle
et des intentions



Développement de story avec Tock Admin



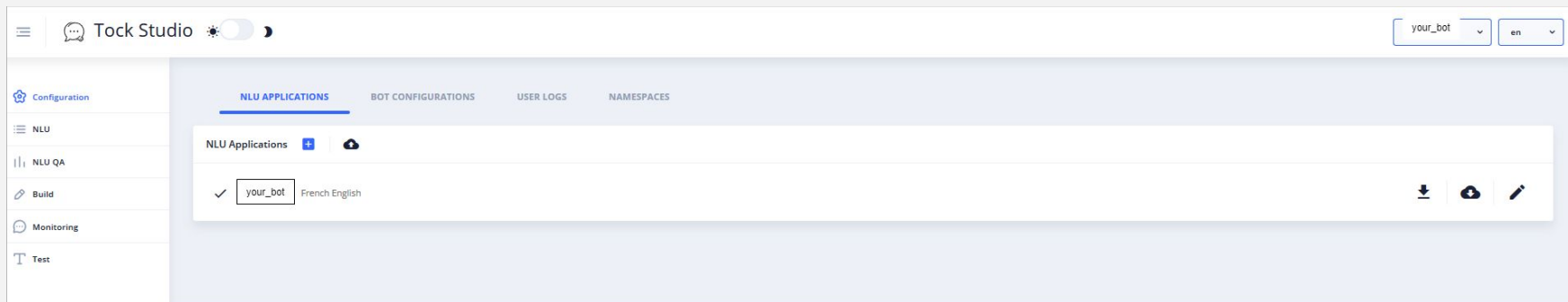
Développement de story avec Tock Admin



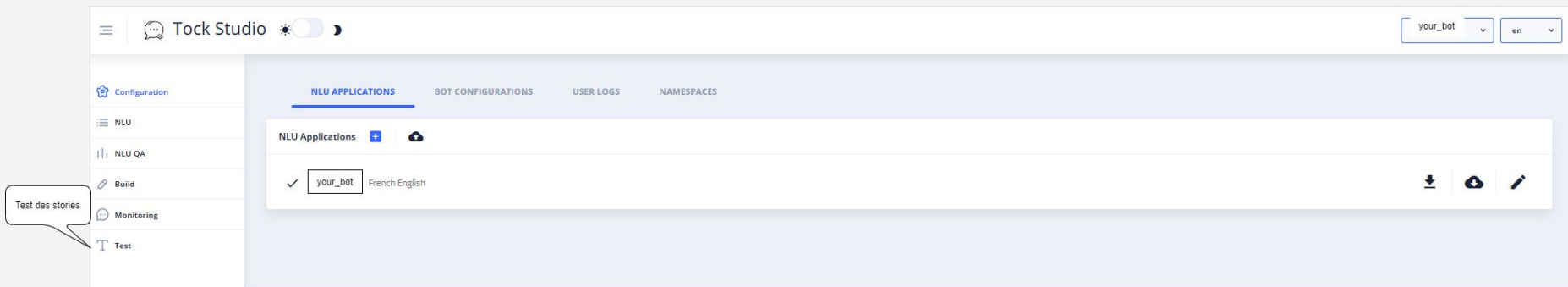
Développement de story avec Tock Admin



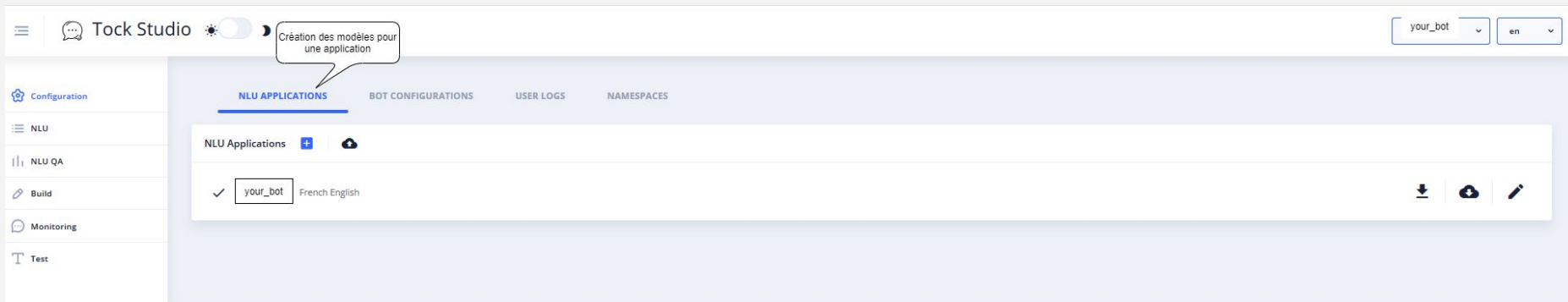
Logs des conversations



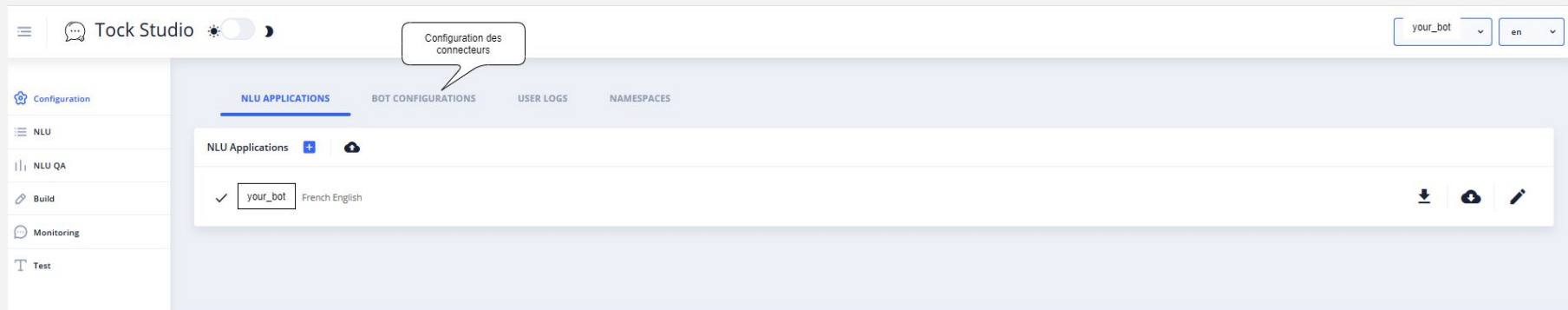
Développement de story avec Tock Admin



Développement de story avec Tock Admin



Développement de story avec Tock Admin



Développement de story avec Tock Admin



Tock Studio

bot_yves en

Configuration

NLU

NLU QA

Build

Monitoring

Test

NLU APPLICATIONS BOT CONFIGURATIONS USER LOGS NAMESPACES

Bot Configurations

Configuration du connecteur Messenger

Connector type: Messenger

Configuration name	Connector identifier	Relative REST path
Connector Custom Configuration		
Application base url	Application id *	
Page id *		
Call Token *		
Webhook token		
Secret *		
Persona id		

CANCEL CREATE

Développement de story avec Tock Admin



Tock Studio

your_bot en

les logs des actions effectuées dans Tock Admin

Configuration

NLU

NLU QA

Build

Monitoring

Test

NLU APPLICATIONS BOT CONFIGURATIONS USER LOGS NAMESPACES

NLU Applications

your_bot French English

Download Upload Edit

Développement de story avec Tock Admin



Tock Studio

Configuration

NLU

NLU QA

Build

Monitoring

Test

TRY IT INBOX UNKNOWN **SEARCH** INTENTS ENTITIES LOGS

Intention à filtrer

Entités à filtrer

Search sentences

booking All destination Entity Roles to exclude Status Modified after Modified before

Advanced Options - Manage Intents and Entities

Items per page: 10 21 - 30 of 780

Je cherche un train direct Metz Marseille

booking French Inbox

origin Metz

destination Marseille

DELETE UNKNOWN VALIDATE ASK FOR REVIEW

14:35:19 2020/01/15

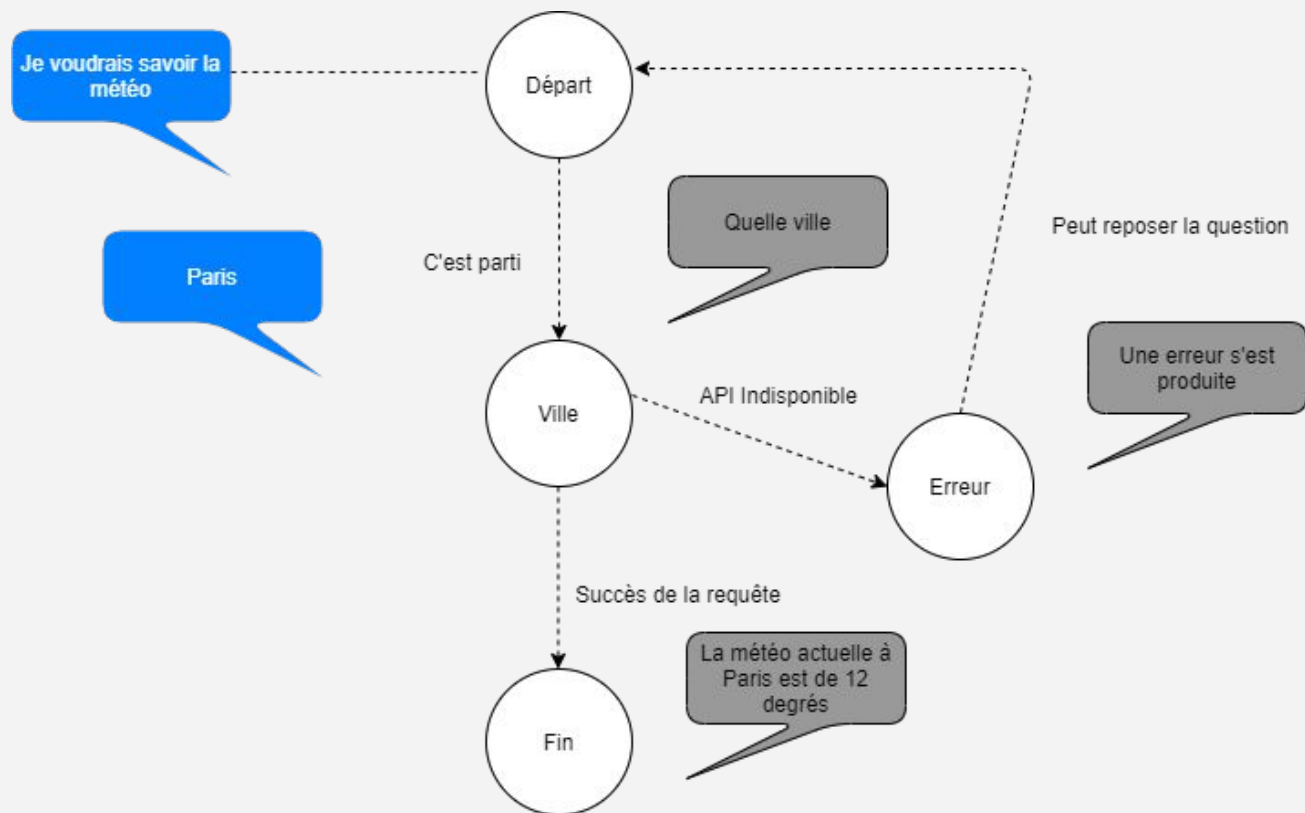
[CANCEL](#) [CREATE STORY](#)

■ Développement de story en Kotlin

- Un IDE
- Repository :
<https://github.com/sydneygael/tock-demo-kaibee>

Développement de story en Kotlin

Automate



À savoir

Développement de story en Kotlin : À savoir

On peut définir des stories avec des étapes :

- `storyWithSteps <SimpleStoryStep>` : story avec gestion des états de l'automate
- `storyDefWithSteps<ConnectorDef, Step>` : story avec gestion des états de l'automate et des comportements des connecteurs
- Possibilité d'avoir des intentions secondaires (indication de nom, prénom etc..)
- voir le reste ici :
<https://doc.tock.ai/tock/fr/dev/bot-integre/>

■ Développement de story en Kotlin : À savoir

En mode Bot intégré :

- Installer la stack sur docker :

<https://github.com/theopenconversationkit/tock-docker>

- Lancer le bot en local

- Il existe un widget customisable et développer en React

<https://github.com/theopenconversationkit/tock-react-kit>

Ressources

Vue générale

<https://doc.tock.ai/fr/>

Documentation

<https://doc.tock.ai/tock/fr/>