

Machine Learning Model Deployment



Sujet 12 (Equipe 2)

- Yahya Rochdi (INSEA)
- Mohamed Essabir (ENSAJ)
- Hamza El Belghiti (FSR)
- Abdelilah Akhmim (ENSAJ)
- EL Qessouar Tariq (ENSET Mohammedia)





ETAT D'AVANCEMENT DU PROJET

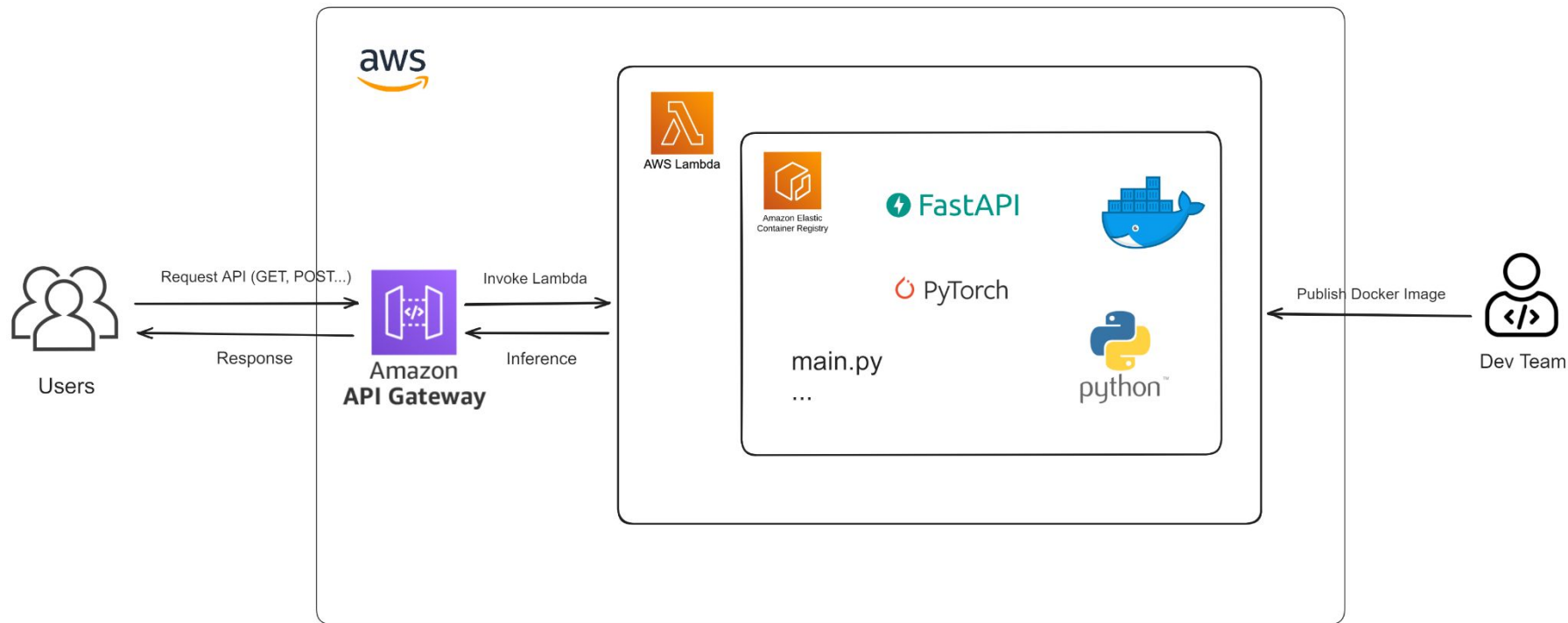
1. Démo de la Version Final
2. Déploiement sur AWS
3. Nouvelles Fonctionnalités Front End
4. Conclusion



DEMO

DÉPLOIEMENT SUR AWS

ARCHITECTURE SUR AWS CLOUD



DÉPLOIEMENT SUR AWS

CONFIGURATION DE LAMBDA

teethsegapi

Throttle Copy ARN Actions

▼ Function overview Info

teethsegapi

API Gateway (3)

+ Add trigger

+ Add destination

Description

-

Last modified

17 seconds ago

Function ARN

arn:aws:lambda:us-east-1:099072753431:function:teethsegapi

Function URL Info

https://zts5x7dqsc4nubg4hzzrotvw40cxlvv.lambda-url-us-east-1.on.aws/

Image Test Monitor Configuration Aliases Versions

General configuration Info Edit

Triggers

Permissions

Destinations

Function URL

Description

-

Timeout

1 min 0 sec

Memory

4096 MB

SnapStart Info

None

Ephemeral storage

512 MB

DÉPLOIEMENT SUR AWS

PRICING DE LAMBDA

Prix x86

Prix (Arm)

Région : USA Est (Virginie du Nord) ▾

Mémoire (Mo)	Tarif par 1 ms
128	0,0000000021 USD
512	0,0000000083 USD
1 024	0,0000000167 USD
1 536	0,0000000250 USD
2 048	0,0000000333 USD
3 072	0,0000000500 USD
4 096	0,0000000667 USD

POUR NOTRE CAS: 0,0013\$/request  1,3\$ pour 1000 request

DÉPLOIEMENT SUR AWS

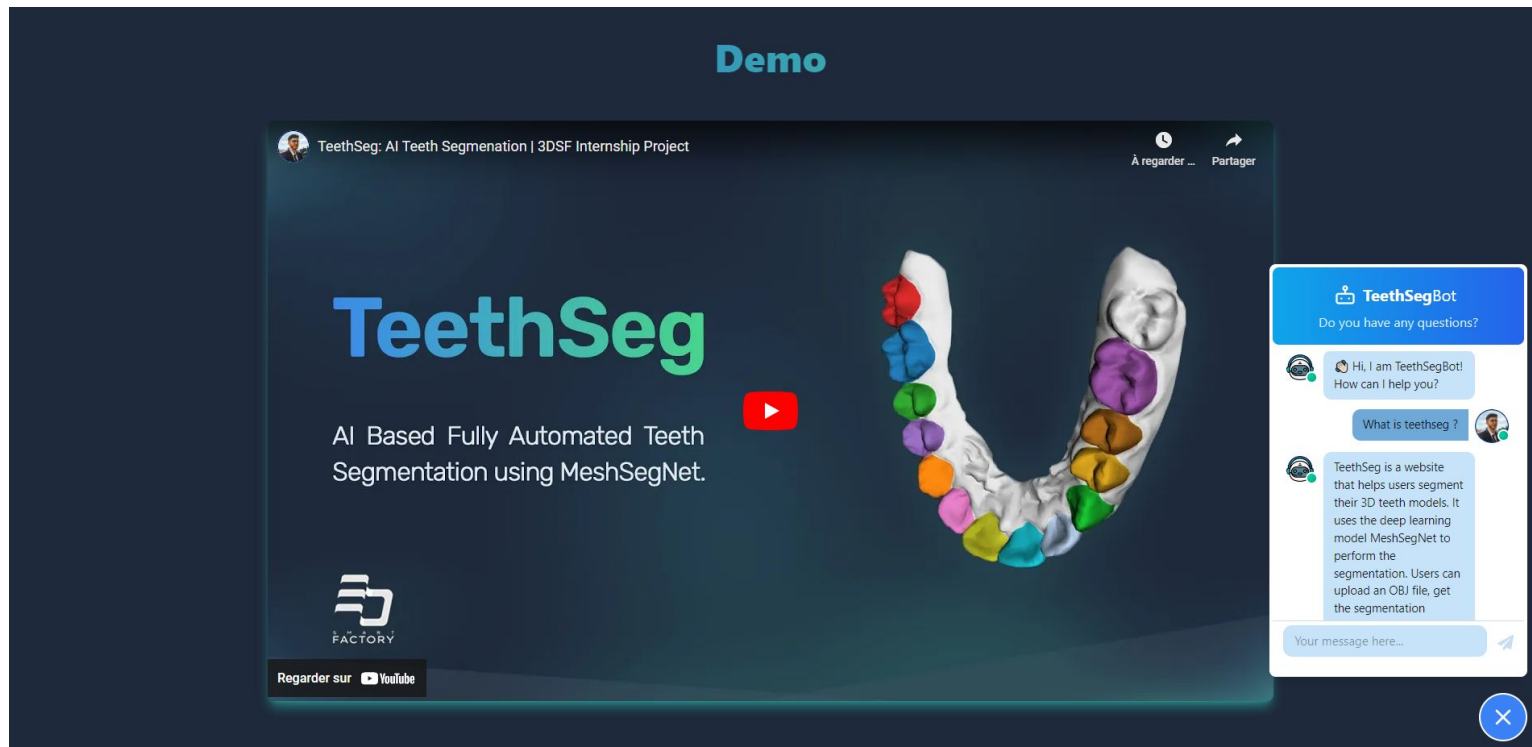
LOGS DE L'API

▶	2023-08-31T00:46:31.457+01:00	START RequestId: 5f048932-7e7f-43af-9485-98a904ac1ea1 Version: \$LATEST
▶	2023-08-31T00:46:31.500+01:00	Predicting Sample filename: Sample_7.obj
▶	2023-08-31T00:46:31.500+01:00	Downsampling...
▶	2023-08-31T00:46:31.521+01:00	Predicting...
▶	2023-08-31T00:46:41.437+01:00	Refining by pygco...
▶	2023-08-31T00:46:46.358+01:00	Upsampling...
▶	2023-08-31T00:46:46.839+01:00	Sample filename: Sample_7.obj completed
▶	2023-08-31T00:46:46.839+01:00	Computing time: 15.34 sec
▶	2023-08-31T00:46:46.877+01:00	END RequestId: 5f048932-7e7f-43af-9485-98a904ac1ea1
▶	2023-08-31T00:46:46.877+01:00	REPORT RequestId: 5f048932-7e7f-43af-9485-98a904ac1ea1 Duration: 15420.11 ms Billed Duration: 15421 ms Memory Size: 4000 MB Max Memory Used: 3965 MB

MEMOIRE UTILISE: 3.8GB

NEW FEATURES

AJOUT D'UN CHATBOT ASSISTANT EN UTILISANT OPENAI API (GPT-3.5-TURBO)



NEW FEATURES

AJOUT DE LA DOCUMENTATION DU FRONT ET BACK END

The screenshot shows the TeethSeg website with a dark blue header. The header contains the TeethSeg logo, navigation links (HOME, DOCS, START, ABOUT, CONTACT), and a Log Out button. Below the header, there are two buttons: Front End Docs and Back End Docs. The Back End Docs button is active, leading to the Back End Documentation section. This section has a 'Download as PDF' button. Below this is the 'I. Back End Structure' section, which contains a paragraph describing the backend architecture and a diagram of the AWS Cloud architecture.

Back End Documentation

[Download as PDF](#)

I. Back End Structure

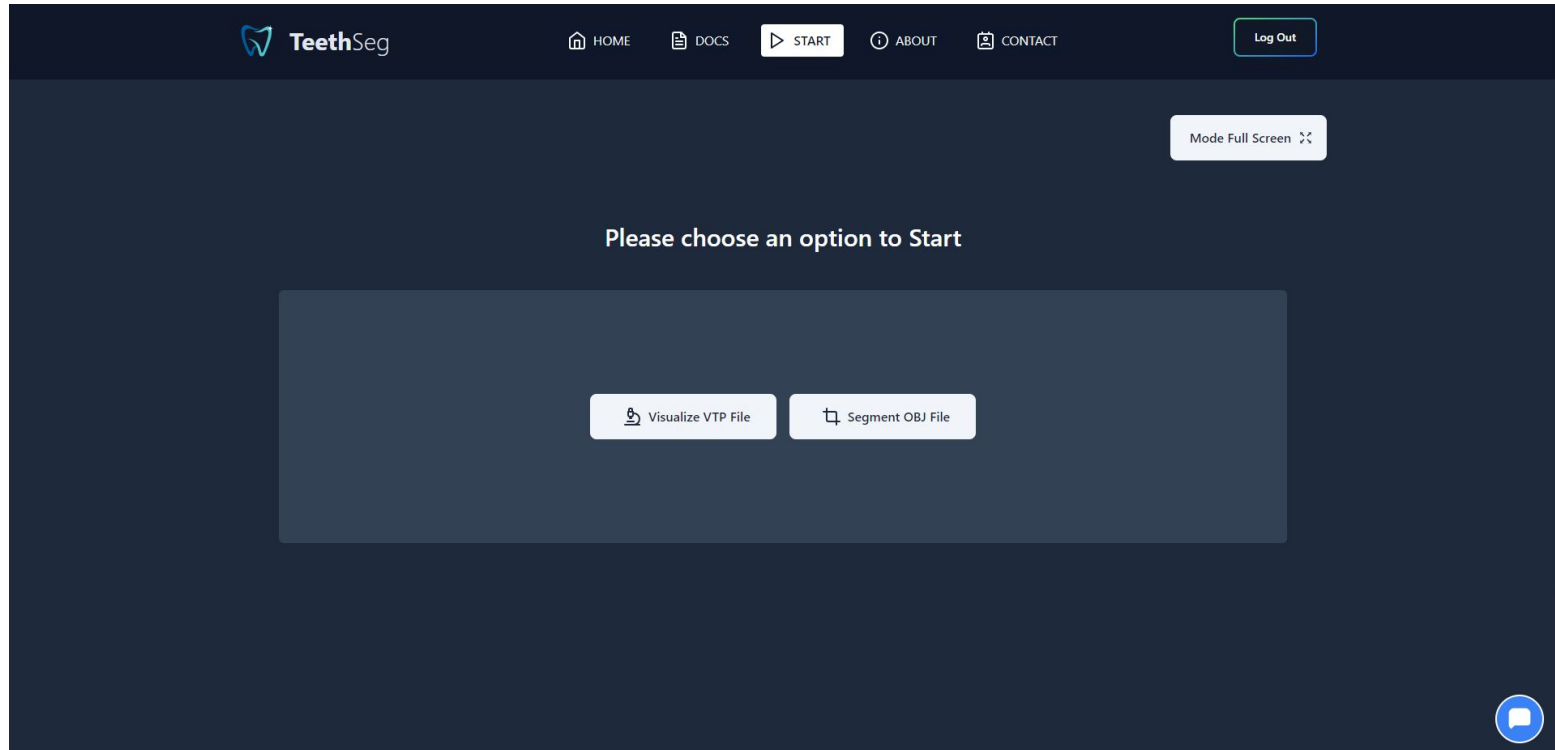
The backend of TeethSeg is a RESTful API developed using Python FastAPI, ensuring robust functionality and reliability. For deployment we used the AWS Cloud using the Amazon Elastic Container Registry (ECR), Lambda and API Gateway services. Here's the architecture of the backend in the AWS Cloud.

The diagram illustrates the AWS Cloud architecture for TeethSeg. It shows the flow from Users to Amazon API Gateway, which then invokes AWS Lambda. The Lambda function is a containerized application running on Amazon Elastic Container Registry (ECR). The application uses FastAPI, PyTorch, and Python. The Dev Team is shown publishing Docker images to ECR. The diagram is divided into three main sections: Users, AWS Cloud, and Dev Team.

```
graph LR
    Users[Users] -- "Request API (GET, POST...)" --> APIGateway[Amazon API Gateway]
    APIGateway -- "Response" --> Users
    APIGateway -- "Invoke Lambda" --> Lambda[AWS Lambda]
    Lambda -- "Inference" --> APIGateway
    subgraph AWS_Cloud [AWS Cloud]
        ECR[Amazon Elastic Container Registry]
        subgraph Container [Container]
            FastAPI[FastAPI]
            PyTorch[PyTorch]
            Python[python]
            main_py[main.py]
        end
    end
    DevTeam[Dev Team] -- "Publish Docker Image" --> ECR
```

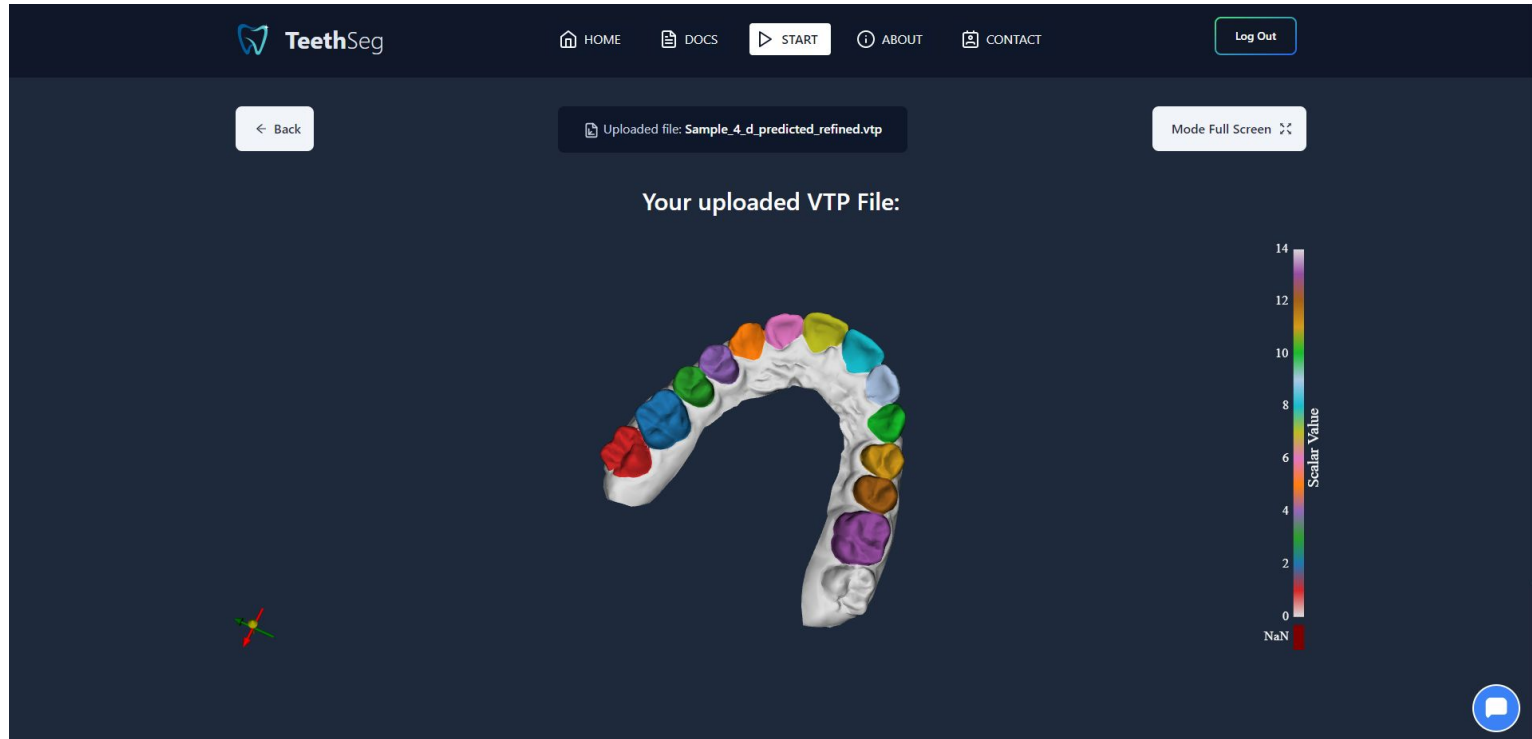
NEW FEATURES

AJOUT DE LA VISUALISATION DES FICHIERS VTP



NEW FEATURES

AJOUT DE LA VISUALISATION DES FICHIERS VTP





CONCLUSION

- Notre parcours au sein de 3D Smart Factory a été une expérience d'apprentissage exceptionnelle, renforçant nos compétences techniques dans la visualisation 3D et notre compréhension du domaine.
- En tant qu'équipe de stagiaires, nous avons développé TeethSeg, une application qui simplifiera la segmentation dentaire en 3D, contribuant ainsi à l'industrie dentaire.
- Nous tenons à remercier nos encadrants qui ont joué un rôle essentiel en nous guidant tout au long du projet, offrant des conseils et une expertise inestimables.