

GOAT

Visualisateur de données génétiques

Plan

- Mise en contexte
- Front-end
- Back-end
- Démonstration
- Recommandations

Mise en contexte



Front-end

Front-end

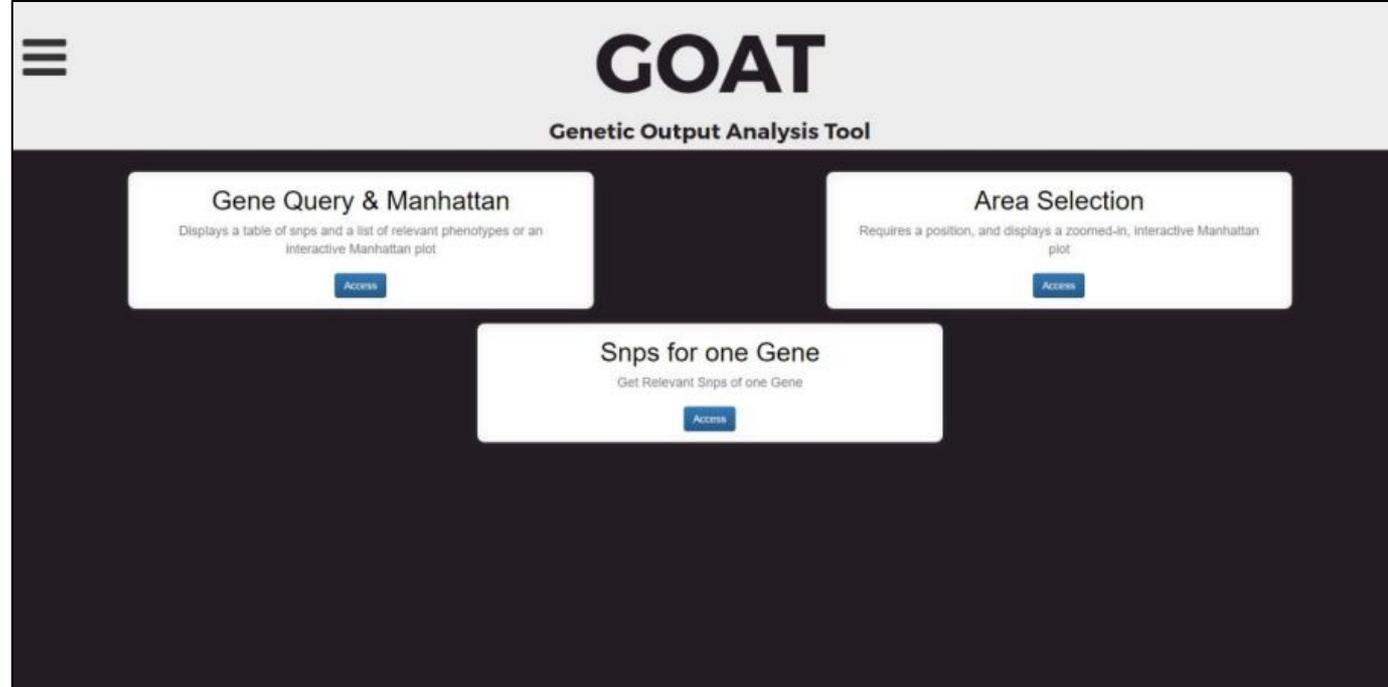
- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



État initial

Trois fonctionnalités déjà présentes dans l'application.

Cependant... avec le départ du Dr. Hamet...



État initial

Area Selection :
entrée de paramètres

The screenshot displays the GOAT (Genetic Output Analysis Tool) interface. At the top, there is a grey header with a hamburger menu icon on the left and the text "GOAT Genetic Output Analysis Tool" in the center. Below the header, the main content area is dark grey. A white modal box titled "Choose your params" is centered on the screen. It contains three input fields: "rsID" with a placeholder "rsID - format rsXXXXX with X between 0 and 9", "Gene" with a placeholder "Gene - Enter the name of the Gene", and "Phenotype" with a placeholder "All cause death". Below these fields are two blue buttons: "Interactive Manhattan" and "Table Data". Below the modal box, there is a "Filter Results" input field and a table of results. The table has a header "Phenotypes" and contains two rows: "All cause death" and "Unmet Renal Needs (uACR)". At the bottom of the table, there is a pagination control showing "1 / 1".

GOAT
Genetic Output Analysis Tool

Choose your params

rsID
rsID - format rsXXXXX with X between 0 and 9

Gene
Gene - Enter the name of the Gene

Phenotype
All cause death

Interactive Manhattan Table Data

Filter Results

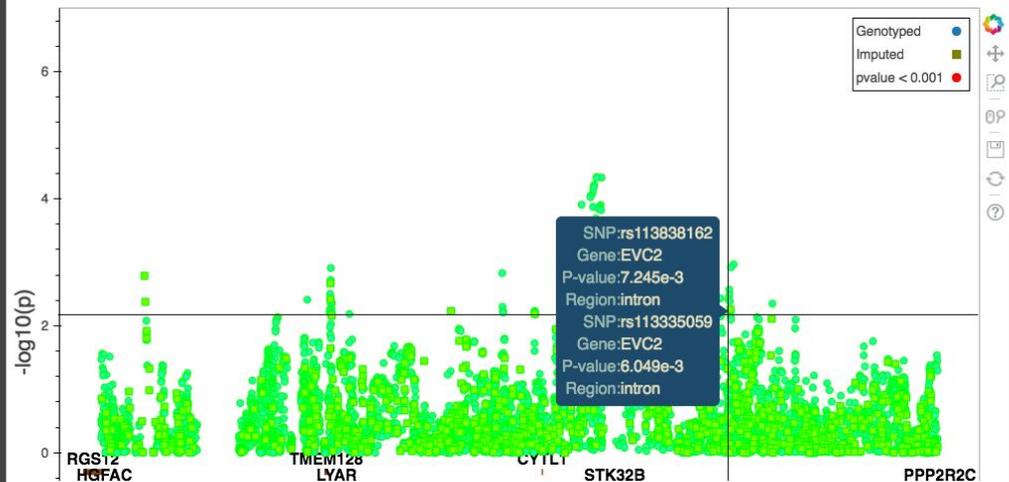
Phenotypes
All cause death
Unmet Renal Needs (uACR)

1 / 1

État initial

Area Selection : Après près de 5 minutes d'attentes...

- Complexité du graphique
- Grande quantité de données
- Légende?
- Interactivité



rs_ID	Chr	P-value	Gene Before	Gene	Gene After	Pos	experiment	info_assoc	Allele A	Allele B	cohort AA	cohort BB	Beta Assoc	maf	all_OR	Covariates	Phenotype	Risk Allele	Risk Af	Risk Allele Beta
rs16943389	15	0.00037721	LINC00928	TICRR	KIF7	90162682	5	1	T	C	2294	71	0.25198	0.15025	1.2937	age sex pc1 pc2	Unmet Renal Needs (uACR)	C	0.1502517306	0.2519800067
rs254244	5	0.00084927				164947504	5	0.90265	A	G	112.93	2179	0.23015	0.17491	1.257	age sex pc1 pc2	Unmet Renal Needs (uACR)	G	0.8250858329	0.2301499993
rs31710	5	0.00023814				164951261	5	0.87434	T	G	100.92	221.7	0.26297	0.16788	1.2856	age sex pc1 pc2	Unmet Renal Needs (uACR)	G	0.8321144566	0.2629700005

État initial

Problème de performance...

- Assoc : 11 millions d'entrées
- Genes : 100 000 entrées
- Marqueurs : 8 millions d'entrées

Area Selection : près de 5 minutes à exécuter en local...



affy_s_rs
assoc
auth_group
auth_group_permissions
auth_permission
auth_user
auth_user_groups
auth_user_user_permiss...
covariate
dataset
django_admin_log
django_content_type
django_migrations
django_session
django_site
experiment
genes
hg_rs_history
marq_gene
marqueurs
marqueurs_bck
person
person_dataset
phen_id_val
phenotypes
previous_gene_hugo
synonym_gene_hugo
users
Web_user

Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Tâches à accomplir : **Étape 1**

A : Téléverser un fichier

Upload a file:
Files supported:
(.vcf, .xl, .csv, .txt)

Upload file

The following columns have been detected:
Please identify columns with the following
drop-down menu which columns to use:

- variant	rs_ID
- pos	position
- chr	chromosome
- all_A	Allele A
- All_B	Allele B
- Gene	Ensembl ID or Gene name

B : Entrer manuellement des ID

Paste your inquiry here and select
type of ID:

- rs ID
- Ensembl ID
- Gene ID

ENSG4568564

ENSG4848484

Tâches à accomplir : Étape 2

- Affichage de 24 chromosomes
- Affichage des correspondances
- Légendes pour filtrer
- Tableau des données
- Téléchargement
- En cliquant sur un bloc jaune...

Select type of mutations:

- INDELS
- Substitutions
- Copy number variations
- Gene Fusion
- Methylation

List of Phenotypes

- Breast cancer
- Lung cancer
- Lymphoma
- Colon cancer
- Lymphoblastic Leukemia
- Myeloid Leukemia

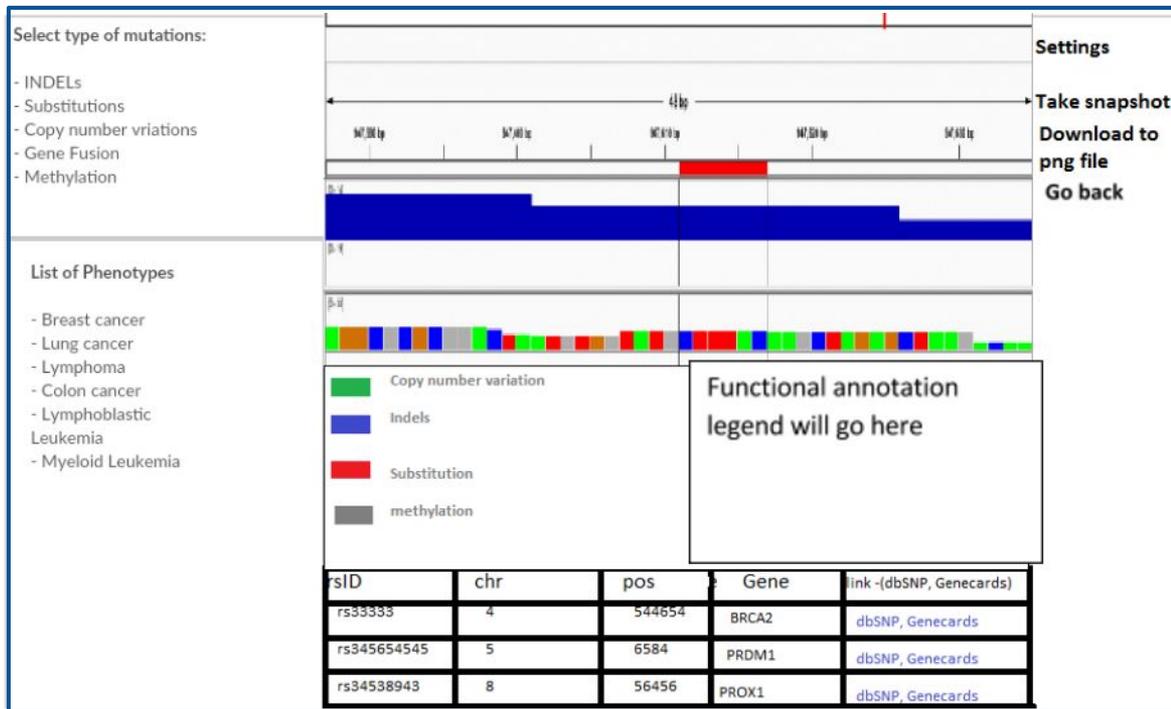
Settings

Take snapshot
Download to png file

rsID	chr	pos	Gene	link -(dbSNP, Genecards)
rs333333	4	544654	BRCA2	dbSNP , Genecards
rs345654545	5	6584	PRDM1	dbSNP , Genecards
rs34538943	8	56456	PROX1	dbSNP , Genecards
rs875039475	10	1564158	C1orf46	dbSNP , Genecards
rs534790874	12	586168	BRCA1	dbSNP , Genecards

Tâches à accomplir : **Étape 3**

- **Interactivité!**
- Détails d'une séquence génétique correspondante



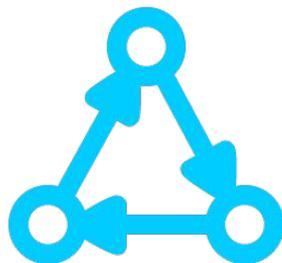
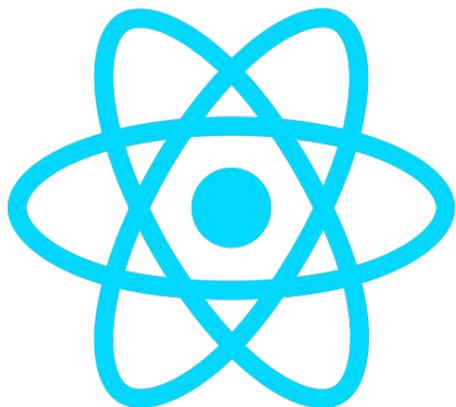
Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Technologies utilisées

django



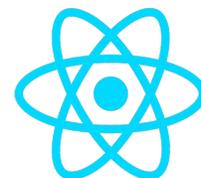
Django

- Utilise le langage de programmation Python
- Cadre Web
- Applications Web dynamiques complètes

The Django logo, featuring the word "django" in a bold, lowercase, sans-serif font. The letter 'j' is stylized with a dot above it. The logo is positioned in the bottom right corner of the slide.

React

- Librairie JavaScript développée par Facebook
- Interfaces utilisateurs interactives
- Single-Page Application
- Composants réutilisables



Reflux

- Librairie JavaScript
- Simplifie le modèle MVC
- *Action* et *Store*



Source: *Github de RefluxJS*



JSX

Préprocesseur qui permet de simplifier la syntaxe utilisée par React

JSX

```
<div className="sidebar"></div>
```

Compilation

React

```
React.createElement(  
  'div',  
  {className: 'sidebar'},  
  null  
)
```



Bokeh

- Librairie Python
- Côté serveur
- Graphiques interactifs
- Capacité à traiter un grand nombre de données



AmCharts

- Librairie JavaScript
- Côté client
- Graphiques dynamiques et interactifs
- Graphiques très personnalisables
- “Responsive”



Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Installation du projet et améliorations

- Difficultés à faire fonctionner le projet
- Installation des librairies nécessaires
 - Fichier requirements.txt manquant
 - Mauvaise version de Bokeh

- Améliorations apportées afin de faciliter le démarrage du projet
 - Documentation
 - requirements.txt
 - Virtualenv

Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Accélérer le processus de développement

- Problèmes de performance

- Base de données MySQL
- Table contenant 8 millions d'entrées



- Solutions

- Tronquer la table contenant les données génétiques utilisées pour Genome Viewer
- 54 entrées

Fonctionnalités Front-end

Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Téléversement d'un fichier

- Permettre à l'utilisateur de téléverser son propre fichier
- Fichier contenant un ensemble de données
- Faire la correspondance entre ses en-têtes et les données attendues
- Permet de visualiser plusieurs données à la fois

Upload file

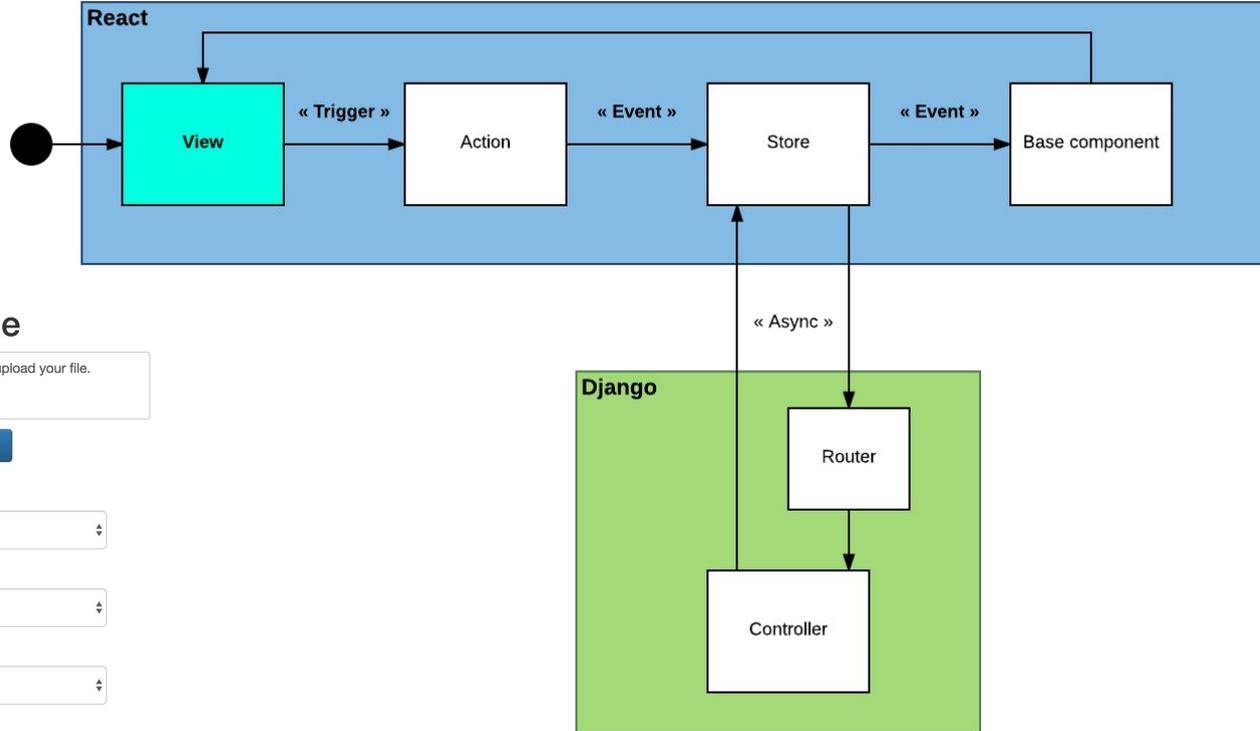
Drag and drop a file or click here to upload your file.

rsld

chromosome

position

Flux de l'application - Extraction d'en-têtes



Upload file

Drag and drop a file or click here to upload your file.

Upload

Submit

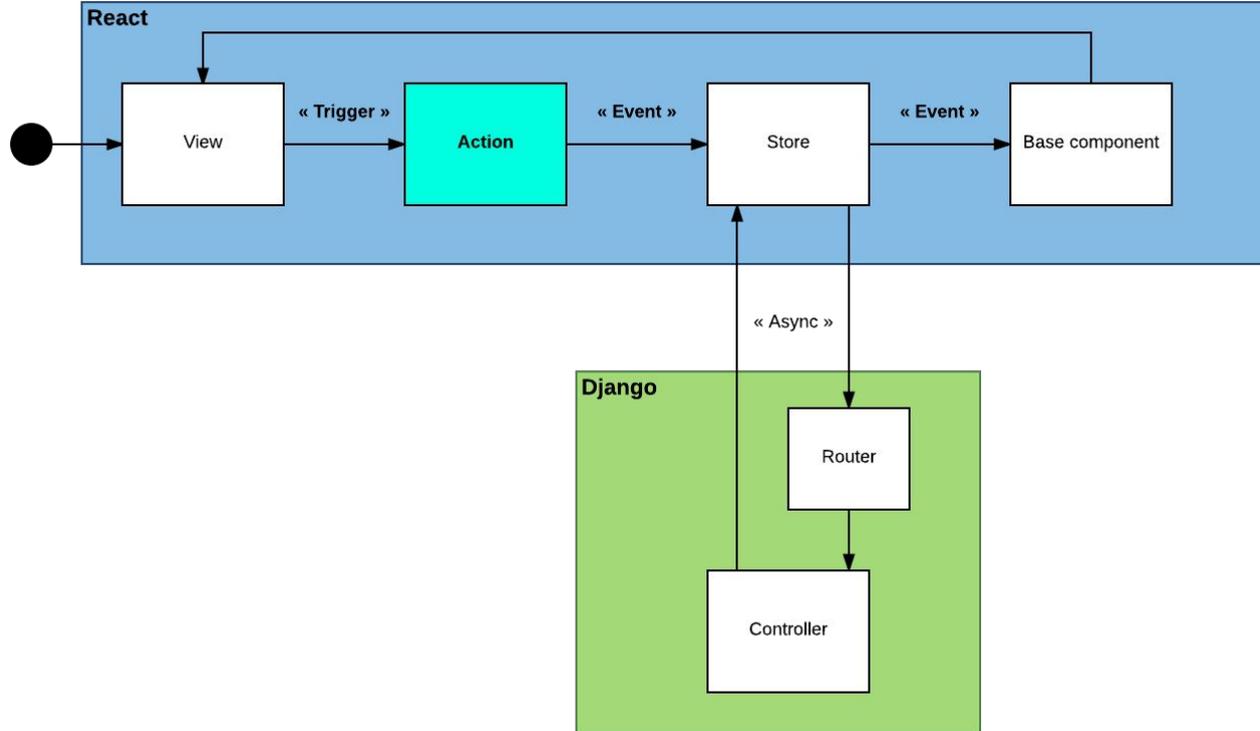
rsld

chromosome

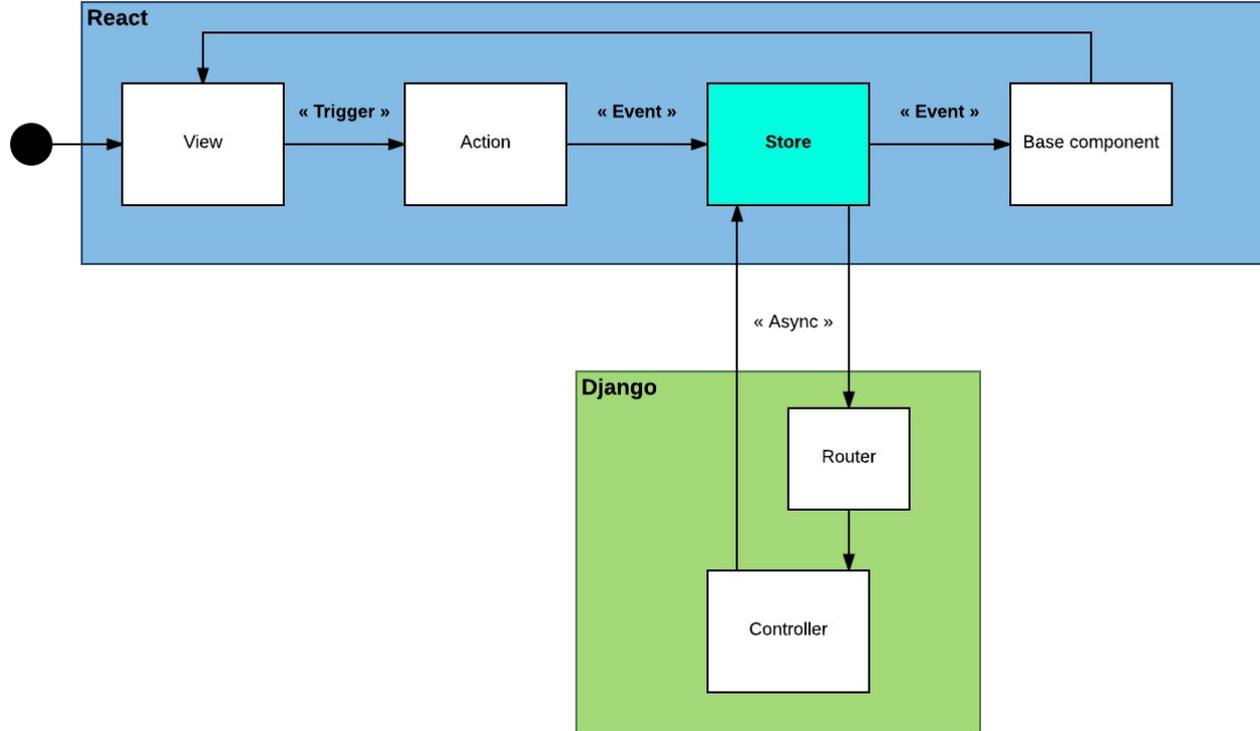
position

Genome Viewer

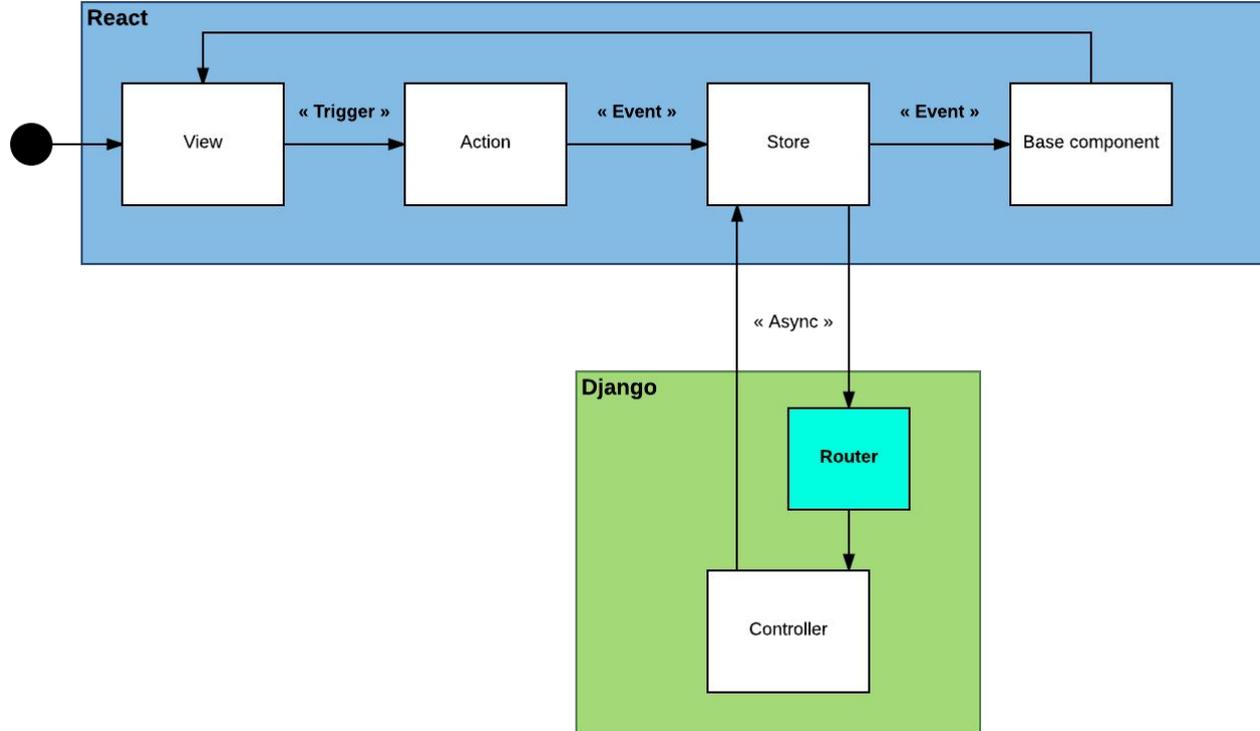
Flux de l'application - Extraction d'en-têtes



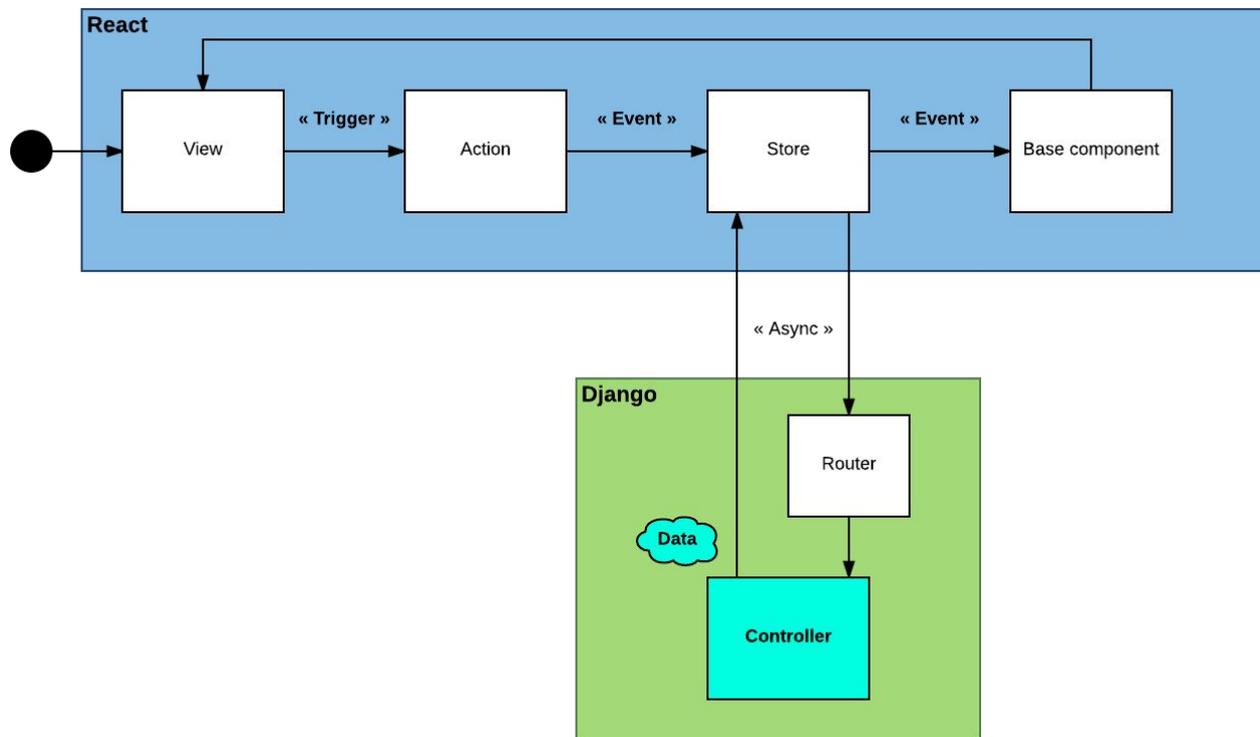
Flux de l'application - Extraction d'en-têtes



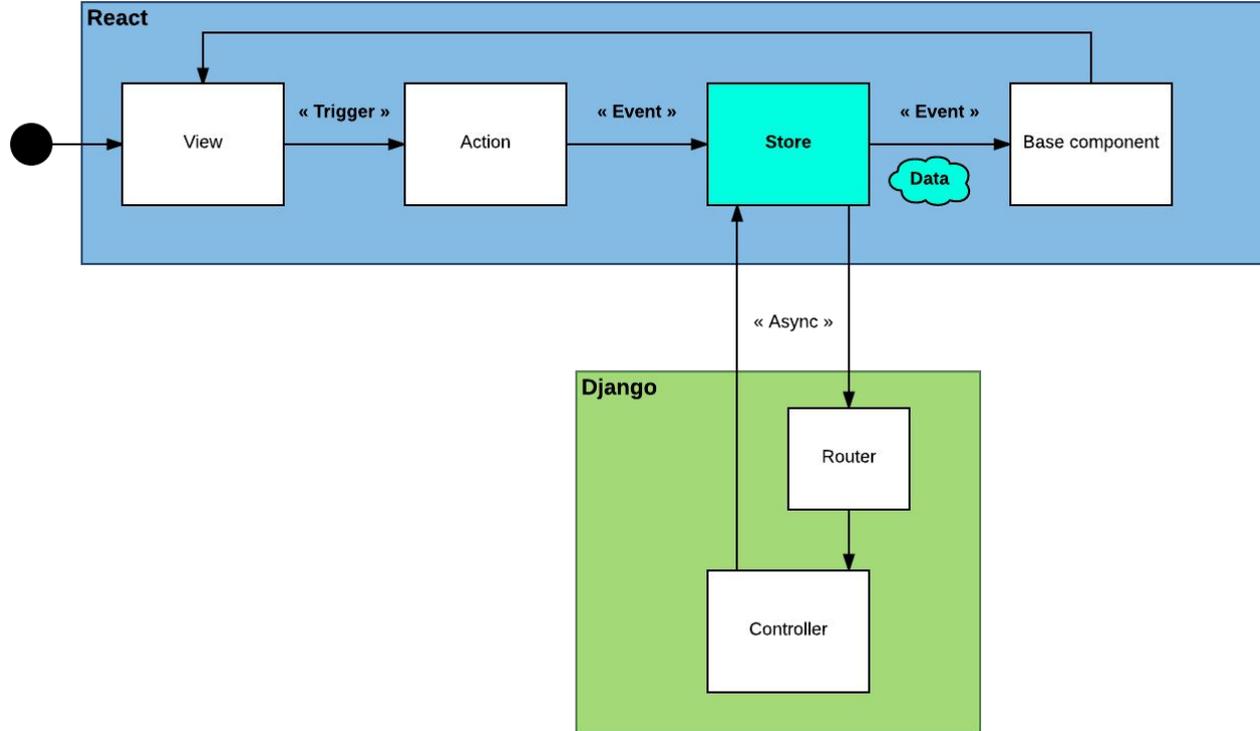
Flux de l'application - Extraction d'en-têtes



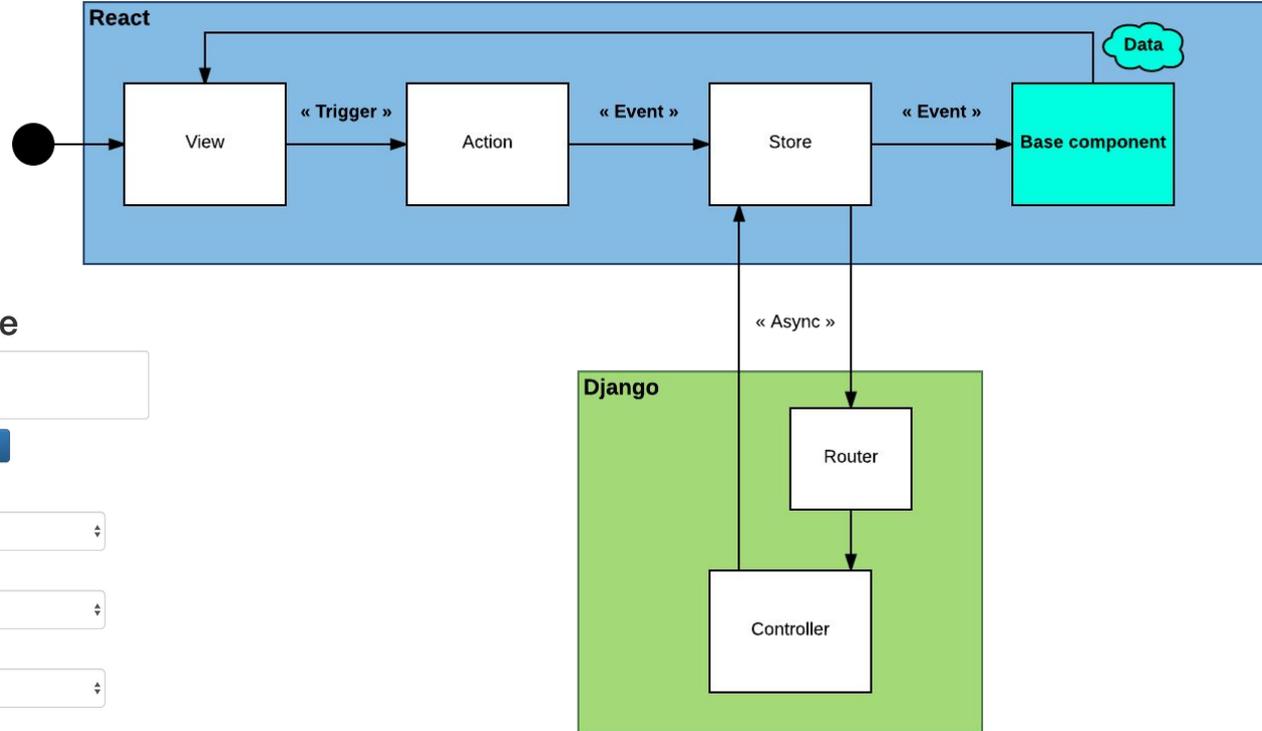
Flux de l'application - Extraction d'en-têtes



Flux de l'application - Extraction d'en-têtes



Flux de l'application - Extraction d'en-têtes



Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Validation de notre compréhension

Correspondance entre
l'interface et la base de
données

Étape 1-A : Téléverser un fichier

- variant	<input type="text" value="rs_ID"/>
- pos	<input type="text" value="position"/>
- chr	<input type="text" value="chromosome"/>



Champs de la table "marqueurs"

```
"idmarqueurs": 1,  
"nom": "rs17095265",  
"sorte": 1,  
"chromosome": 1,  
"position": 74872533,  
"build_id": 19,  
"remapped_from_hg18": "rs17095265",  
"refNCBI": "C",  
"observed": "C/T",  
"classe": "single",  
"func": "intron",  
"frame": "",  
"codons": "",  
"peptides": "",  
"gene": "TNNI3K",  
"gene_strand": "+",  
"start_gen": 74701070,  
"end_gen": 75010116,  
"gene_before": "FPGT",  
"gene_before_strand": "+",  
"dist_gen_before": 198146,  
"start_gen_before": 74663895,  
"end_gen_before": 74674386,  
"gene_after": "ERICH3",  
"gene_after_strand": "-",  
"dist_gen_after": 161262,  
"start_gen_after": 75033794,  
"end_gen_after": 75139422,  
"idgenes": null
```

Validation de notre compréhension

Correspondance entre
les données et le
graphique

Champs d'intérêts

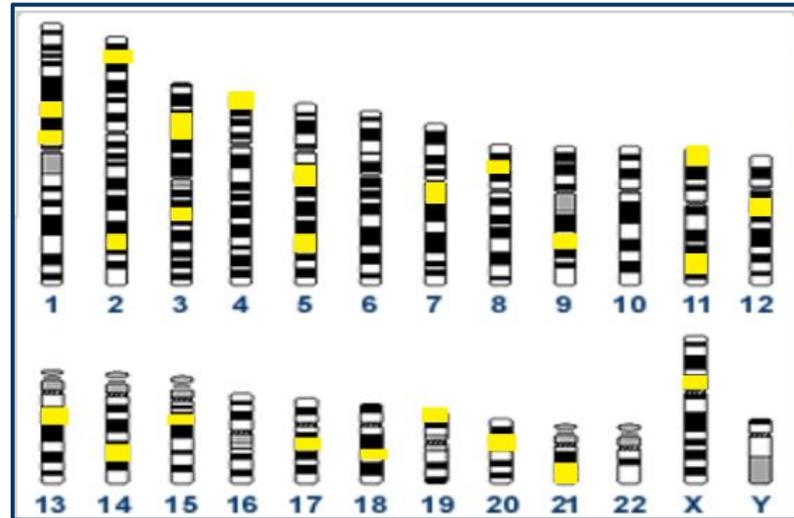
Chromosomes

Nom (rsID)

Position



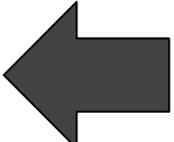
Étape 2 : Affichage des séquences



Validation de notre compréhension

Première tentative :
représenter la séquence
rs2558128

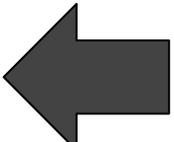
Trouver la
"longueur" d'un
chromosome



```
7  
8 SELECT chromosome, MIN(position) as min, MAX(position) as max FROM marqueurs GROUP BY chromosome;
```

chromosome	min	max
0	195	155239436
1	37965	249222325
2	10587	243185679
3	60197	197897481
4	27955	190939665

Trouver les détails d'un
séquence génétique
correspondante (rsID,
position et chromosome)



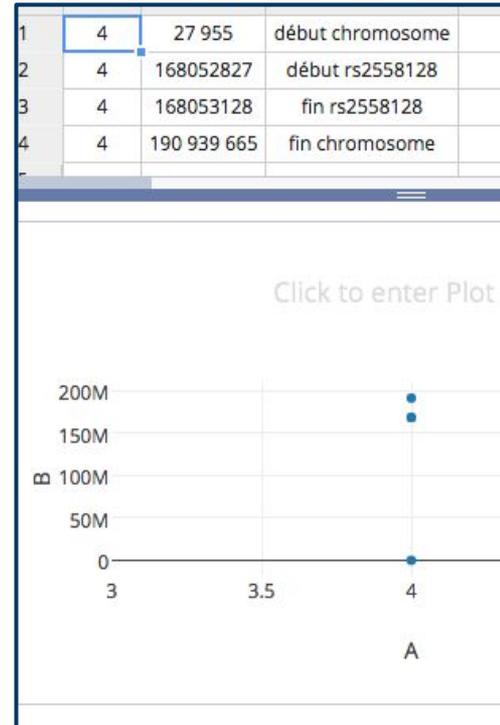
```
3 SELECT * FROM marqueurs  
4 WHERE nom="rs2558128"  
5 AND position=16852827  
6 AND chromosome=4  
7  
8
```

nom	end_position
rs2309396	168053128

Validation de notre compréhension

Première tentative :
représenter la séquence
rs2558128

Affichage très primitive des
données trouvées



Validation de notre compréhension

Première tentative :
représenter la séquence
rs2558128



Validation réussie!

Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts



Graphique avec Bokeh

Code de référence

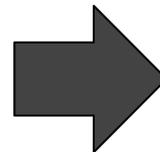
Area Selection



Requête “longueur des chromosomes”

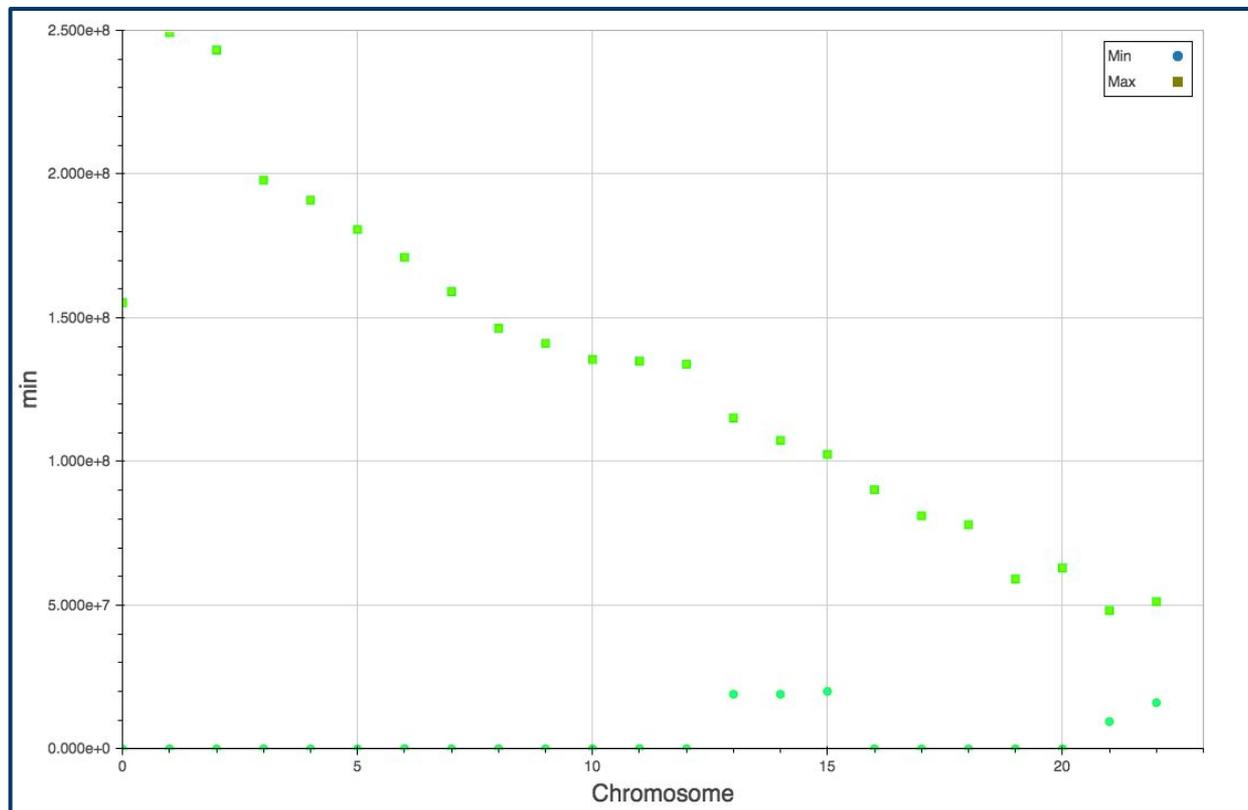
```
7
8 SELECT chromosome, MIN(position) as min, MAX(position) as max FROM marqueurs GROUP BY chromosome;
```

chromosome	min	max
0	195	155239436
1	37965	24922325
2	10587	243185679
3	60197	197897481
4	27955	190939665



...

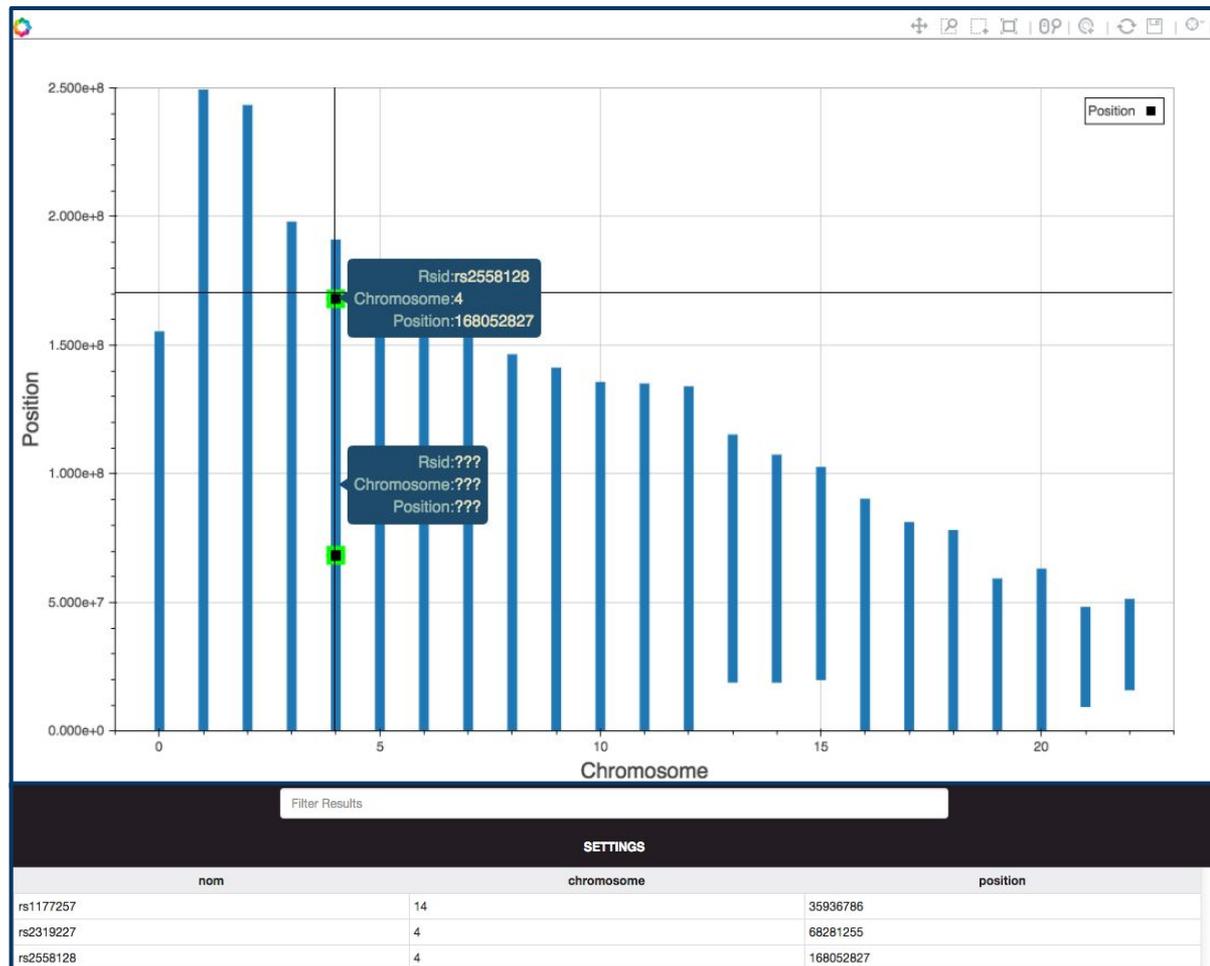
Graphique avec Bokeh



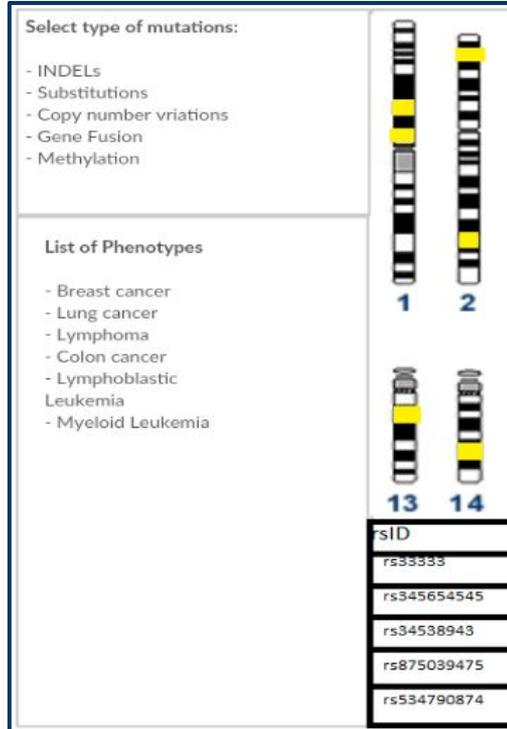
Positions minimales
et maximales des
chromosomes

Graphique avec Bokeh

- Affichage des chromosomes
- Affichage des séquences correspondantes
- Affichage de détails
- Affichage du tableau



Graphique avec Bokeh



La suite :

- Ajouter de l'interactivité
- Filtrer les données affichées
- Malheureusement...



Problème **majeur**

Front-end

- État initial (GOAT v3)
- Tâches à accomplir
- Technologies utilisées
- Installation du projet et améliorations
- Accélérer le processus de développement
- Téléversement d'un fichier
- Validation de notre compréhension
- Graphique avec Bokeh
- Graphique avec AmCharts

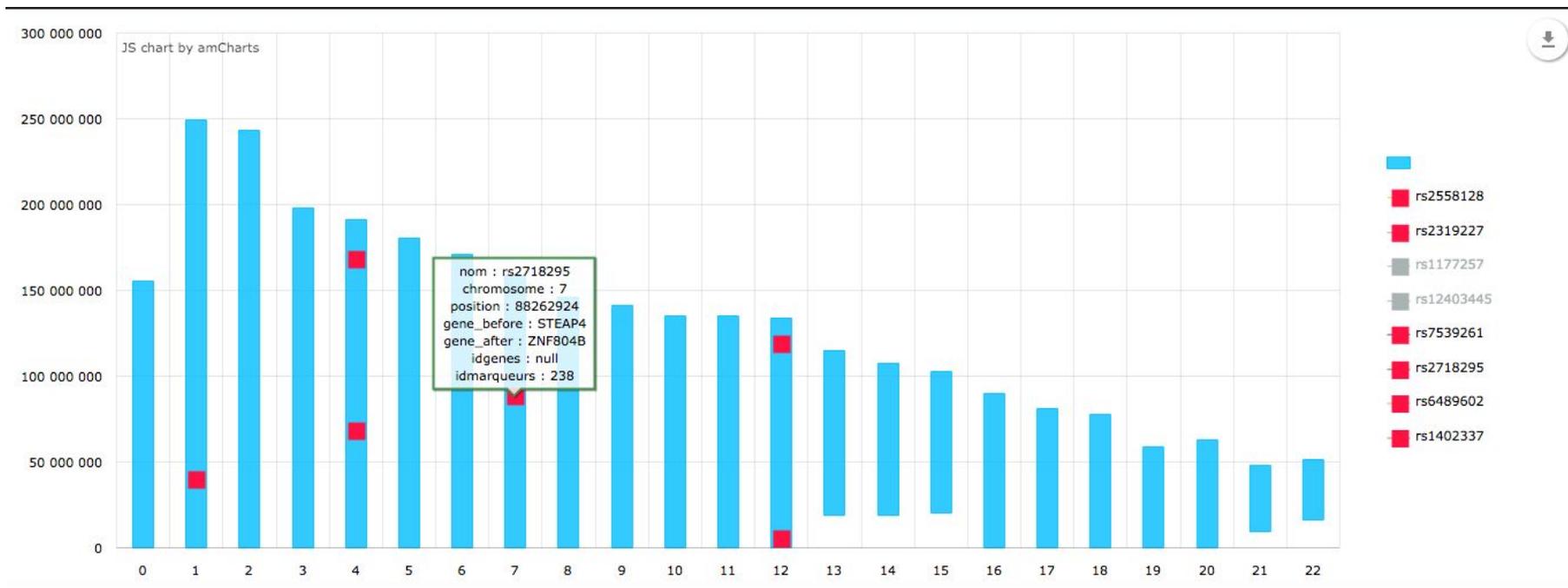


Graphique avec AmCharts.js

On jete et on recommence!

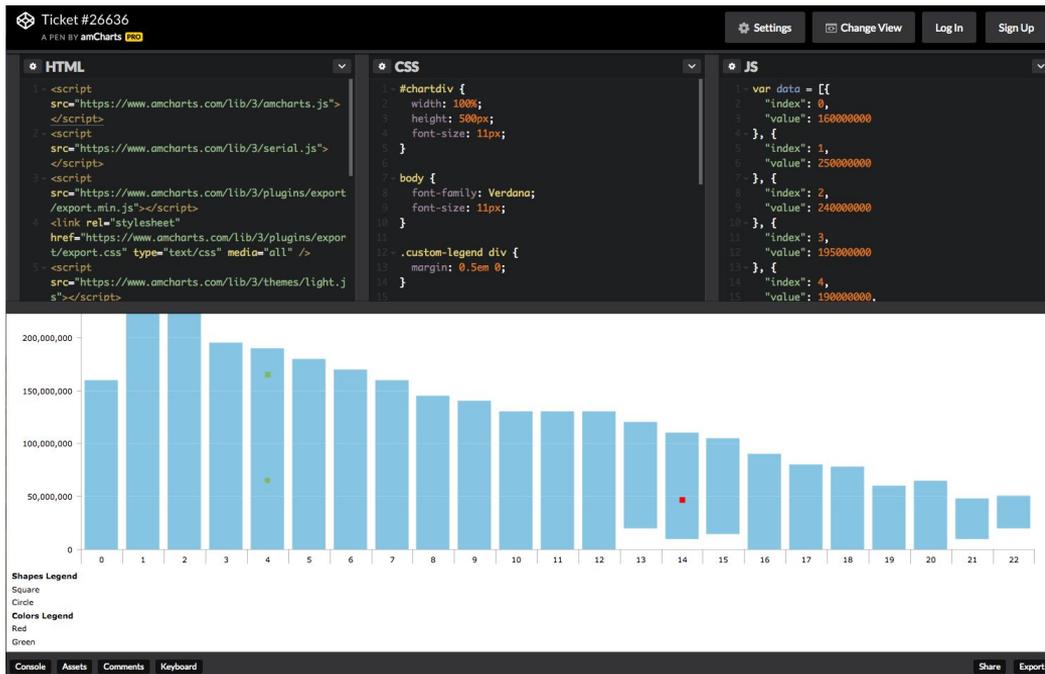
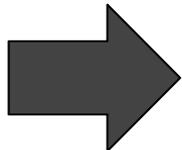
- Installation de la nouvelle librairie
- Changement important au niveau de la manipulation des données
- Beaucoup plus facile à manipuler
- Meilleure documentation

Graphique avec AmCharts.js



Graphique avec AmCharts.js

Demande de support à l'équipe de AmCharts par Béatriz



Graphique avec AmCharts.js

Exemple fourni par AmCharts

```
var chart = AmCharts.makeChart("chartdiv", {  
  "type": "serial",  
  "theme": "light",  
  "dataProvider": data,  
  ...  
  ...  
chart.addListener("init", function onInit (e)  
{  
  ...  
}
```



Notre code : AmCharts.React

```
var reactChart = React.createElement(AmCharts.React, {  
  "type": "serial",  
  "theme": "light",  
  "dataProvider": data,  
  ...  
  ...  
reactChart.addListener("init", function onInit (e) {
```

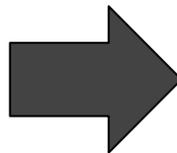
Graphique avec AmCharts.js

Notre code final

```
var chart = AmCharts.charts[0]

var config = {
  "type": "serial",
  "theme": "light",
  ...
}

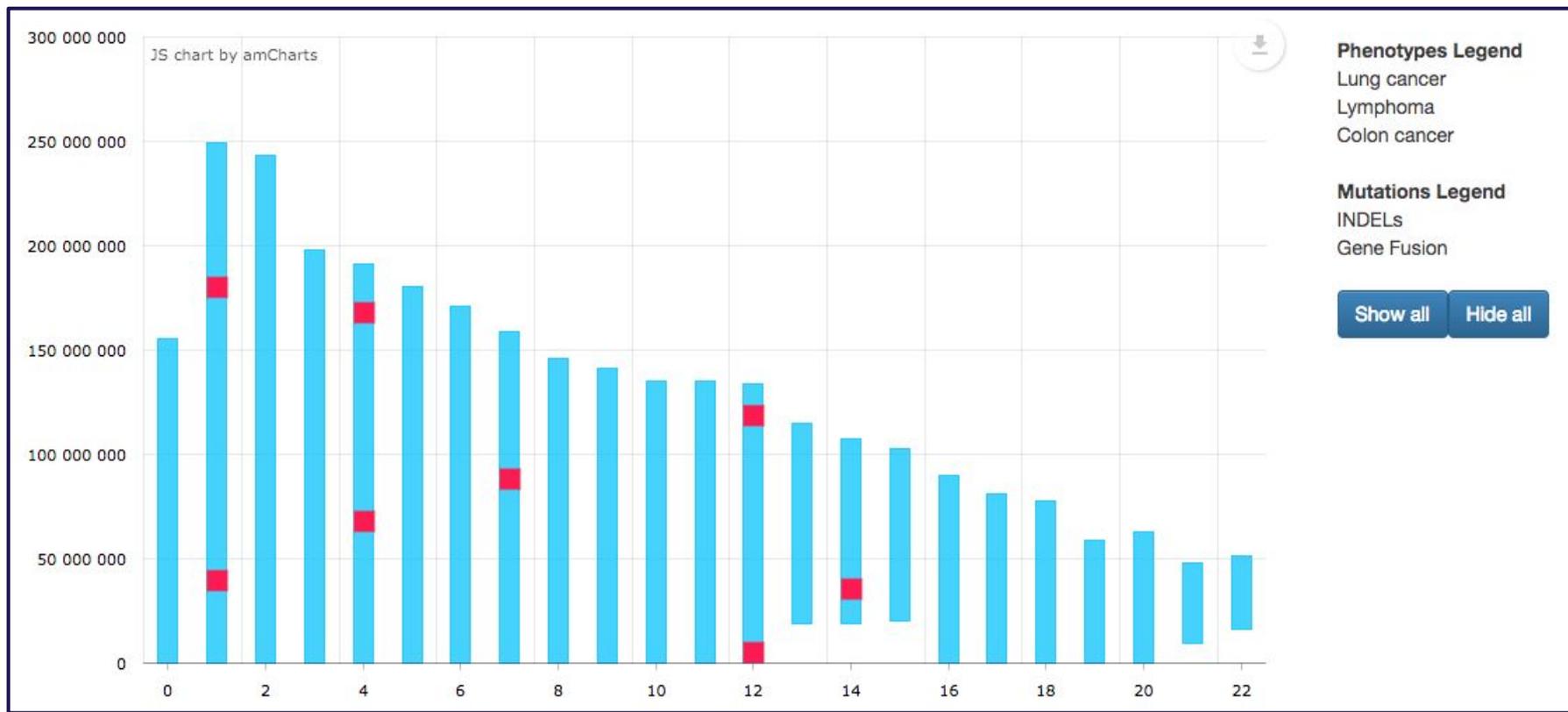
return (
  <div className="container">
    <AmCharts.React {...config} /></div>
  ...
)
```



Derniers ajustements

- Simulation des filtres
"Phénotype" et "Mutation"
- Ajustement dans le code

Graphique avec AmCharts.js



Back-end

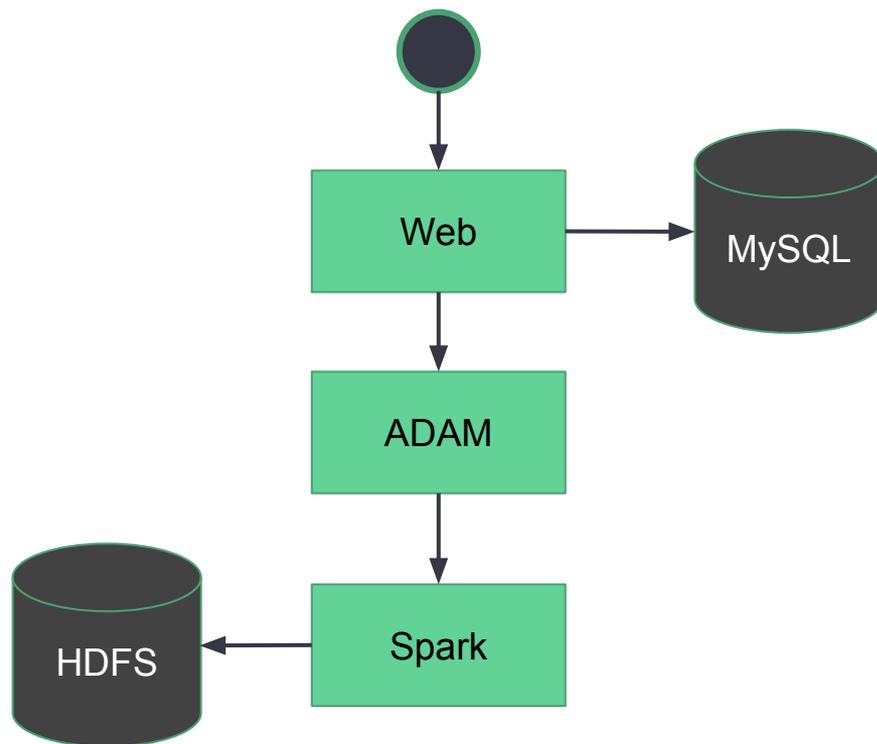
Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
- Difficultés rencontrées
- Comparaison avant / après
- À venir



Solution retenue

- ADAM
 - Plate-forme d'analyse de la génétique avec des formats de fichiers spécialisés.
- 1000 Genomes
 - ADN de plus de 2 500 personnes issues de 26 populations différentes.
- AWS
 - Amazon Web Service

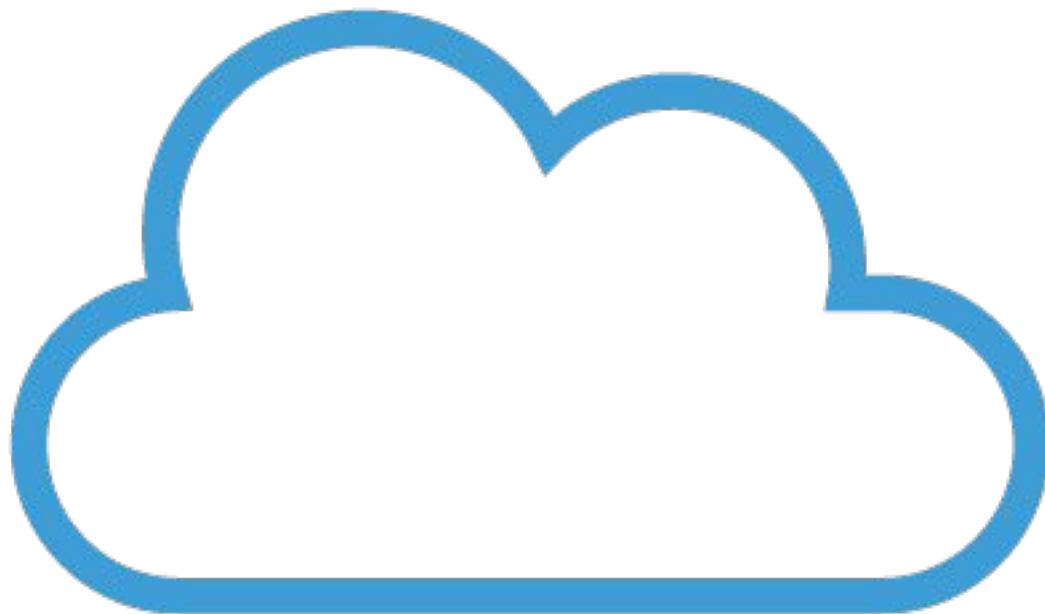


Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
- Difficultés rencontrées
- Comparaison avant / après
- À venir



Objectifs : Infonuagique



Objectifs : NoSQL



Objectifs : Rapidité



Objectifs : Documentation

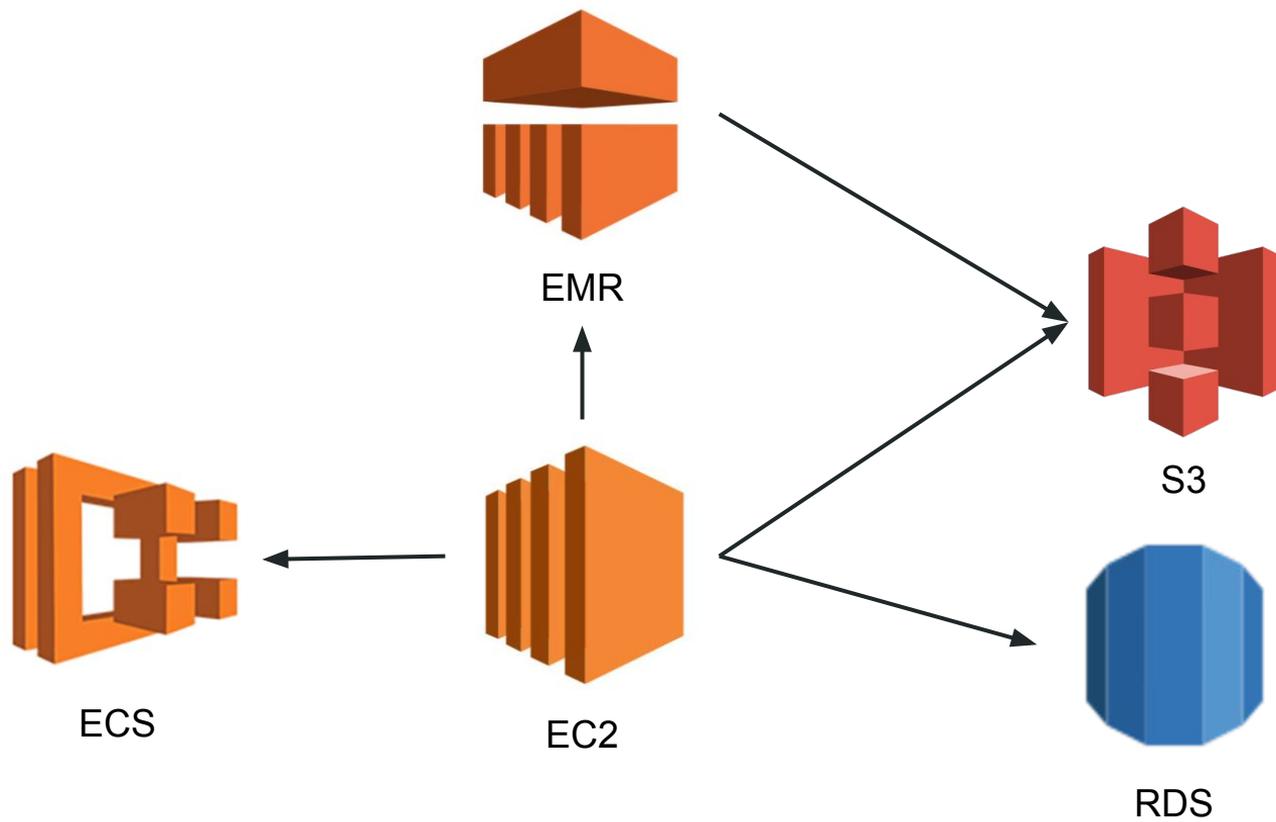


Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
 - AWS
- Difficultés rencontrées
- Comparaison avant / après
- À venir

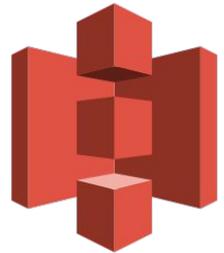


AWS



AWS - S3

- 1000 Genomes
- 30 GB compressé
- 1.5 TB décompressé
- “Bucket” 1000 Genomes gratuit



AWS - EMR

- Hadoop
 - Framework Open Source de *Apache Software Foundation* permettant le développement simple d'applications distribuées.

- Spark
 - Système de traitement de données sophistiqué et conçu pour sa rapidité et sa facilité d'utilisation.



AWS - EC2

- EC2
 - Machine disponible dans le Cloud
 - Contrôle total
 - Services flexibles
 - Démarrage facile et rapide



AWS - Coûts

- S3
 - **1000 Genomes**
 - Gratuit
 - <https://aws.amazon.com/fr/1000genomes/>
 - **Fichier ADAM**
 - Prix: 0.023/GB (100GB = 2.3\$)
- EMR
 - **Type:** m4.large
 - **Prix par mois:** 22\$ (Utilisation à 100%)
- EC2
 - **Type:** m4.large
 - **Prix par mois:** 86\$ (Utilisation à 100%)

Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
 - Django
- Difficultés rencontrées
- Comparaison avant / après
- À venir



Django

- Refonte complète du back-end
- Utilisation du framework ADAM
- Scripts de déploiement

The Django logo, featuring the word "django" in a bold, lowercase, sans-serif font. The letter 'j' is stylized with a dot above it. The logo is positioned in the bottom right corner of the slide.

Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
 - ADAM
- Difficultés rencontrées
- Comparaison avant / après
- À venir



ADAM

- Développement d'une nouvelle fonctionnalité dans ADAM

```
./adam-submit matching [GENOTYPE_FILE] [RSID]
```

- Déploiement de la nouvelle version sur Amazon

Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
- Difficultés rencontrées
- Comparaison avant / après
- À venir



Difficultés rencontrées



ADAM

VCF2ADAM

Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
- Difficultés rencontrées
- Comparaison avant / après
- À venir



Comparaison

- Conversion VCF
 - Sans EMR : 6 heures
 - Avec EMR : 30 minutes
- Requêtes au back-end
 - Sans ADAM : ~3 minutes
 - Avec ADAM : ~5 secondes
- Disponibilité
 - Sans Amazon : Localement seulement
 - Avec Amazon : Accessible de partout

Back-end

- Explication de la solution retenue
- Objectifs
- Travail accompli
- Difficultés rencontrées
- Comparaison avant / après
- À venir



À venir

- Automatisation du déploiement avec ECS
- Contrôle des accès avec une base de données RDS
- Créer un nouveau *bucket* S3 avec les fichiers de 1000 Genomes en format ADAM directement
- Ajouts de fonctionnalités

Démonstration

goat.com

Conclusion
