

# ROLLER COASTER 3000

## Simulation Physique de montagnes russes

Arthur Biot

Anouchka Daniel

Fabien Roy

Clément Bonizec

Martin Graffin

Antoine Vallade

[https://gitlab-cw3.centralesupelec.fr/martin.graffin/roller\\_coaster\\_3000](https://gitlab-cw3.centralesupelec.fr/martin.graffin/roller_coaster_3000)



01

02

03

04

05

## IChoix du Projet

Différents sujets ont été proposés puis nous avons votés pour le meilleur

## Minimal Viable Product

Mise en place des premiers objectifs à réaliser pour obtenir une montagne russe

## Premières Améliorations

La simulation devient interactive

## Simulations Finale

L'utilisateur contrôle maintenant plusieurs paramètres

## Améliorations envisageables

# Choix du Projet

**Idée Pendule double**

**Idée Système Solaire**

**Idée Montagne Russes**

**Idée Modélisation Voiture**

# Choix du Projet

## Construction d'un rail

- Le rail est créé par la méthode B-spline. : seuls les deux points précédant un nouveau point influent sur le tracé de la courbe.
- Le rail est fixé par le code : l'utilisateur ne le choisit pas.
- Le rail comporte une partie qui tracte le wagon (en rouge).

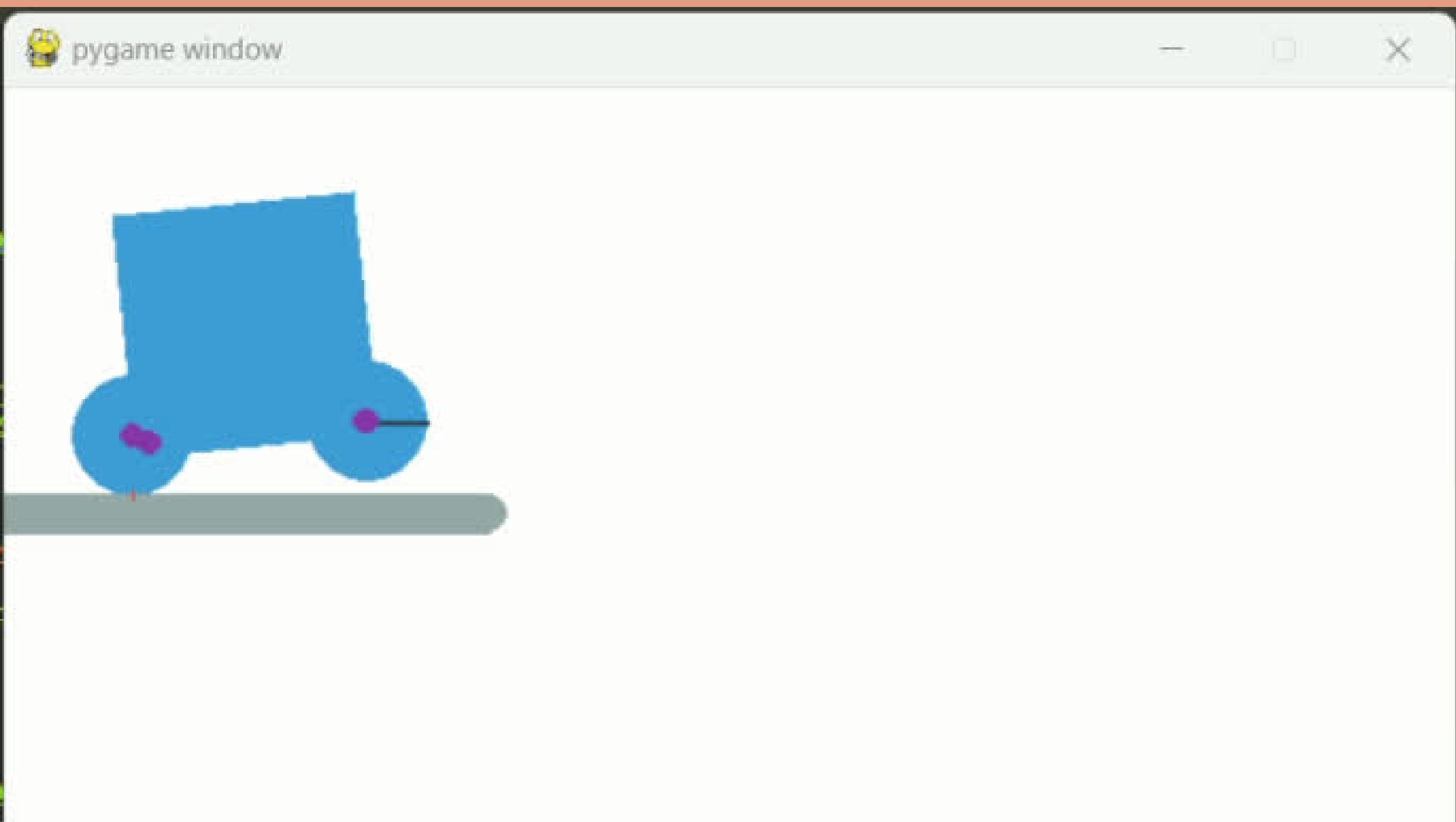
# **Minimal Viable Product**

# Construction d'un wagon

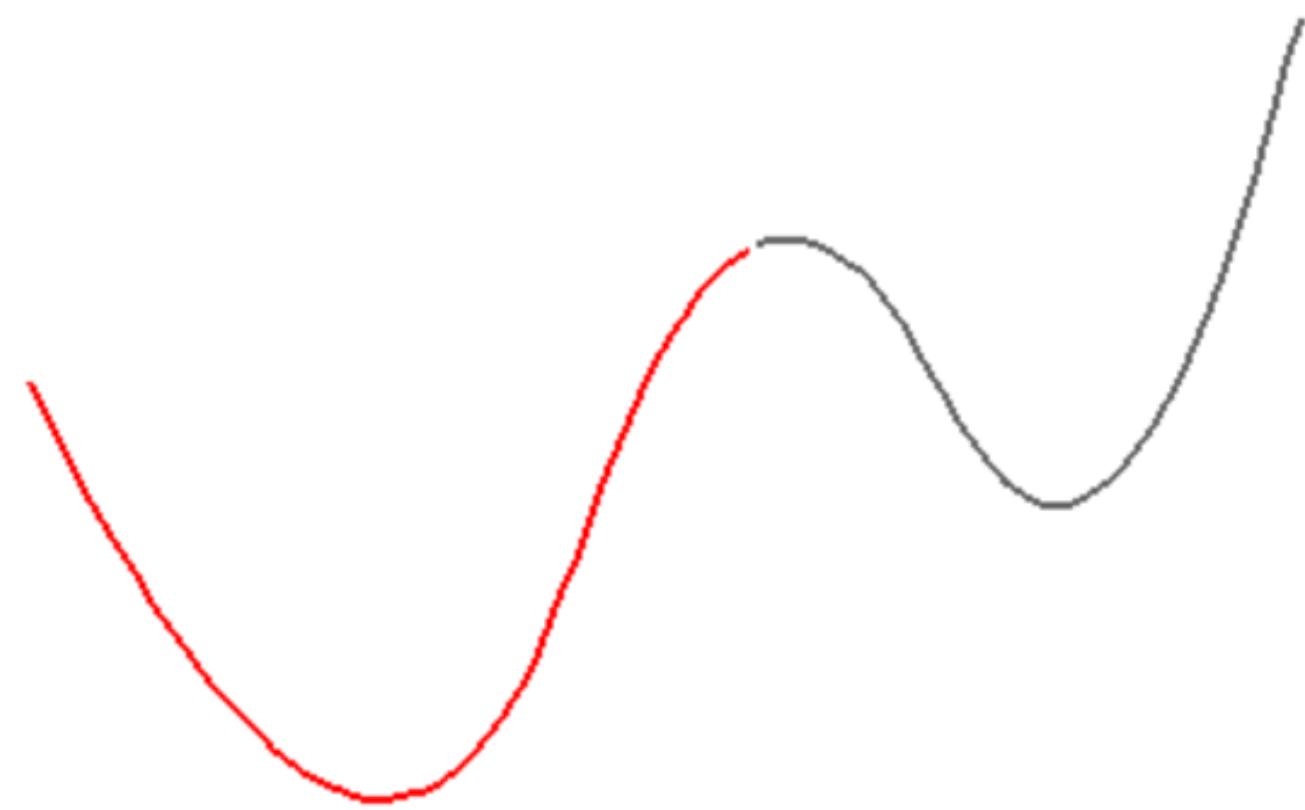
- Le wagon possède un corps et des roues.

- Le wagon doit pouvoir se déplacer sur le rail sans le quitter
- Le wagon est, à l'état initial, sur le rail et sans inclinaison

Arthur -Fabien



Minimal Viable Product



## Construction d'un rail

- Le rail est créé par la méthode B-spline. : seuls les deux points précédant un nouveau point influent sur le tracé de la courbe.
- Le rail est fixé par le code : l'utilisateur ne le choisit pas.
- Le rail comporte une partie qui tracte le wagon (en rouge).

Minimal Viable Product

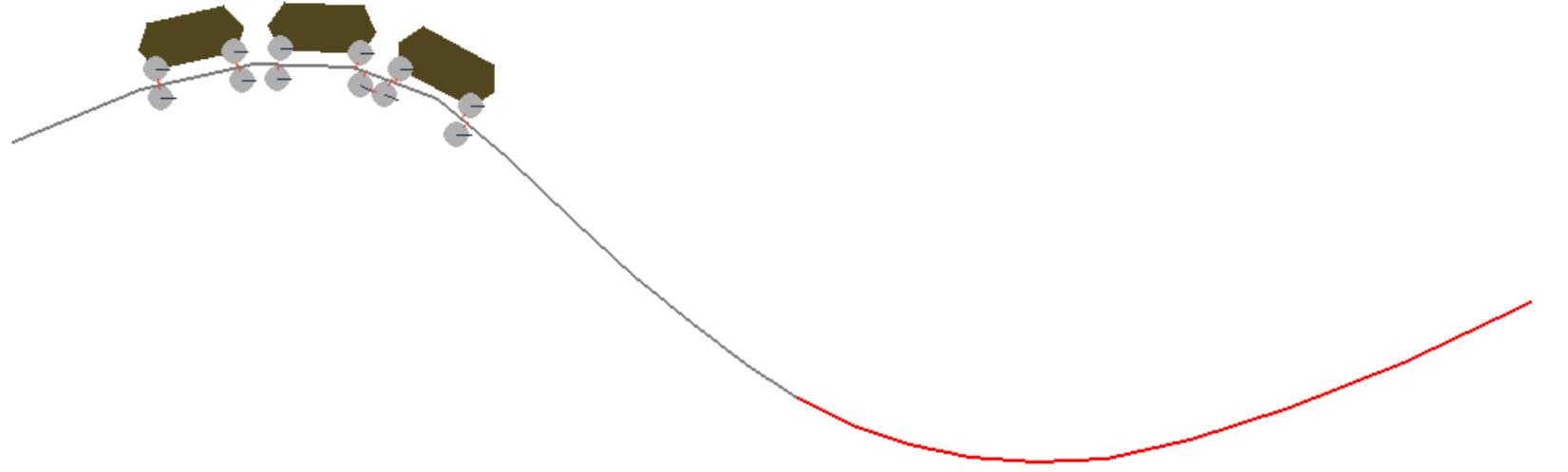
Anouchka - Antoine

# Interface Tkinter

Martin - Clément

Minimal Viable Product

# Premières Améliorations



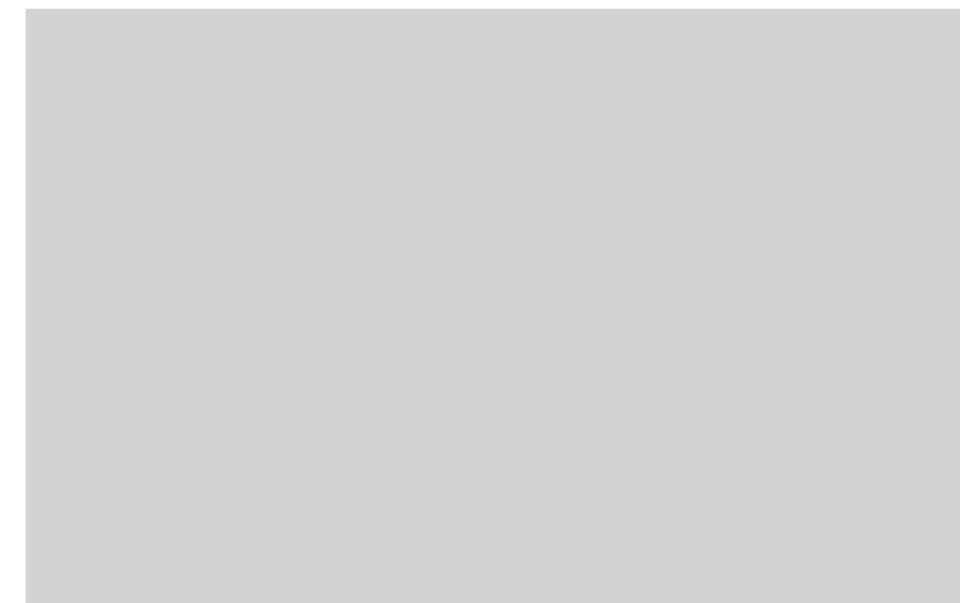
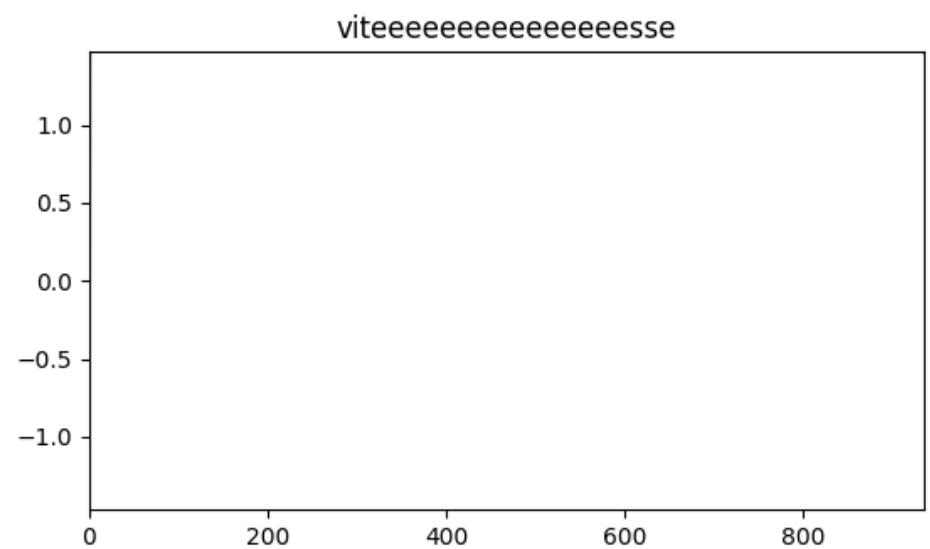
Choix du graphe à afficher

Aucun graphe

Vitesse(t)

Accélération(t)

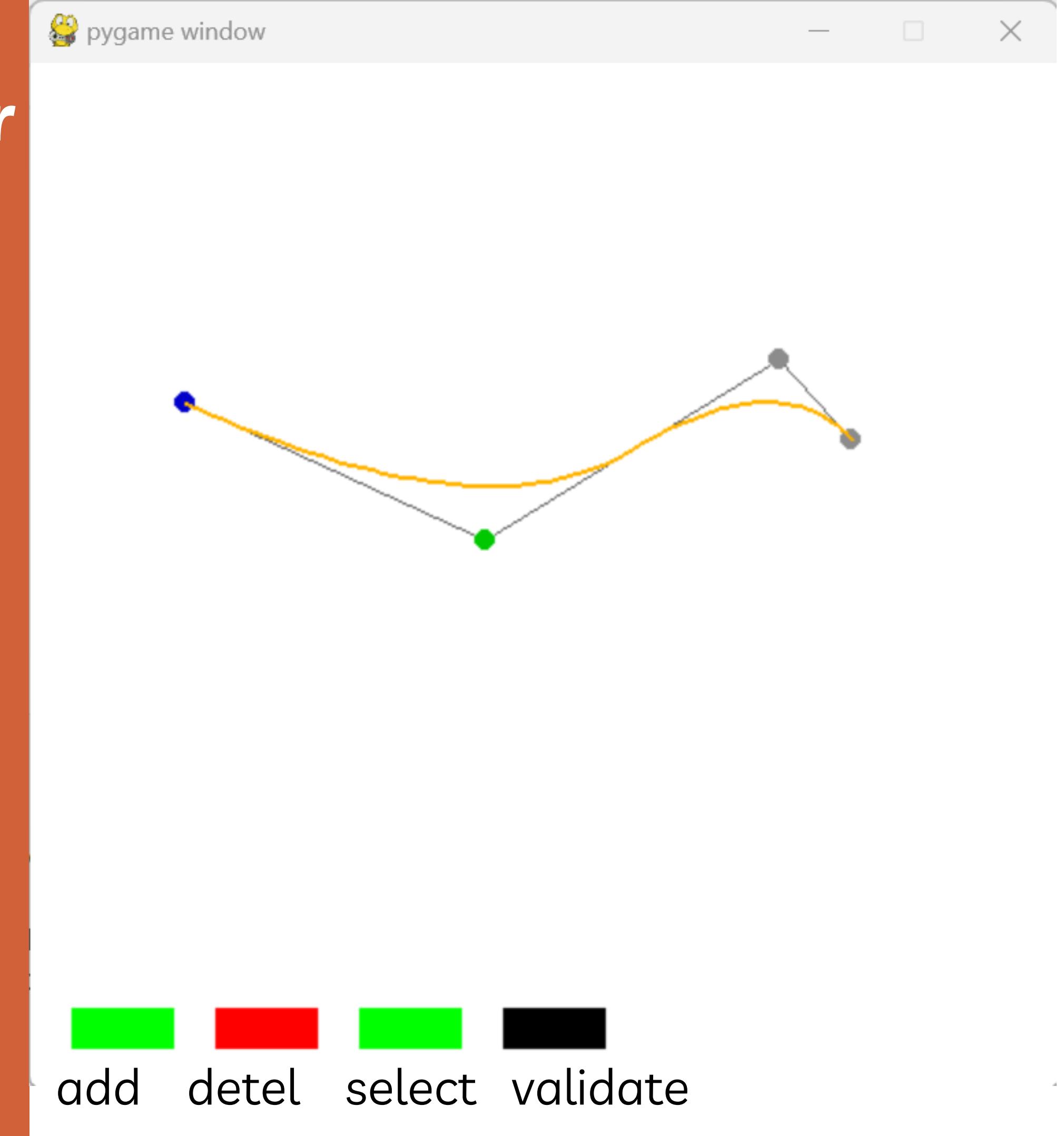
Durée de mesure (s): 10



## Un rail construit par l'utilisateur

- Méthode de B-spline.
- Interface animée Pygame.
- Problème avec la pente de début du rail : il faut qu'elle soit nulle.

Anouchka



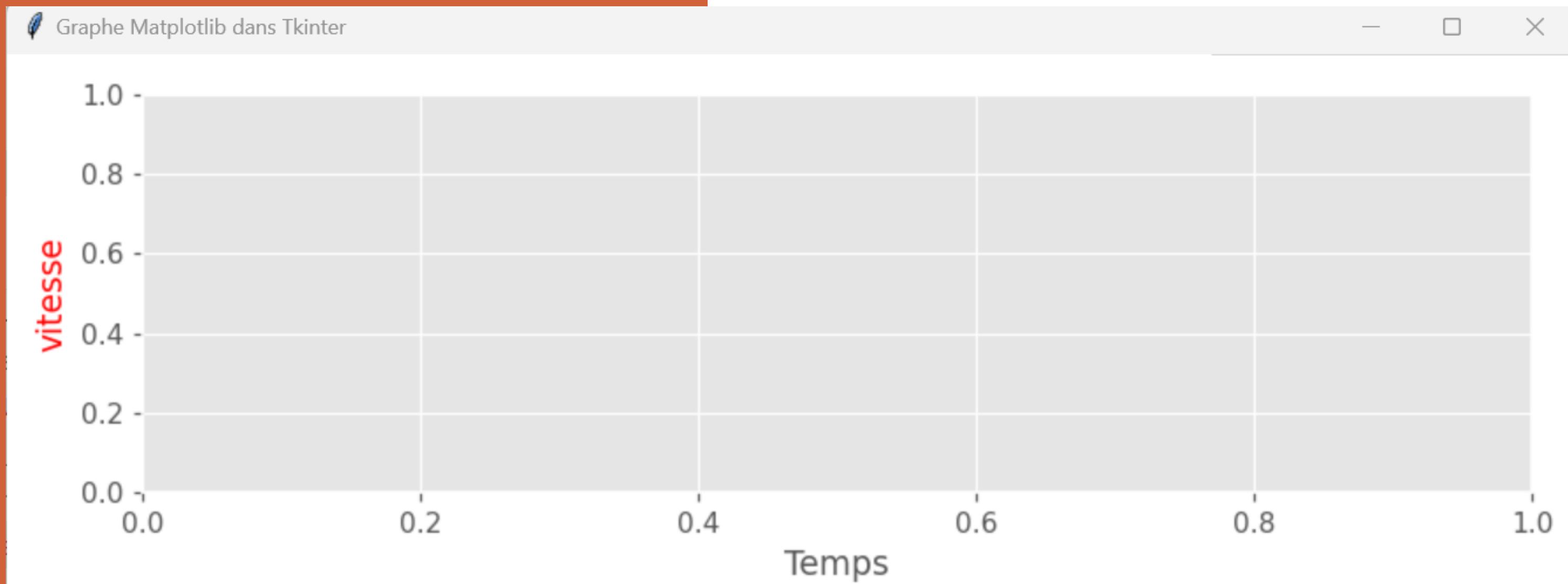
## Un wagon plus stable sur le rail

- Changement de la disposition des roues
- La liaison entre les roues rends la structure plus stable

Arthur - Fabien

## Acquisition de données physiques sur le wagon

- On peut obtenir la vitesse du wagon au cours du temps.



# Premières Améliorations

## Interface Tkinter

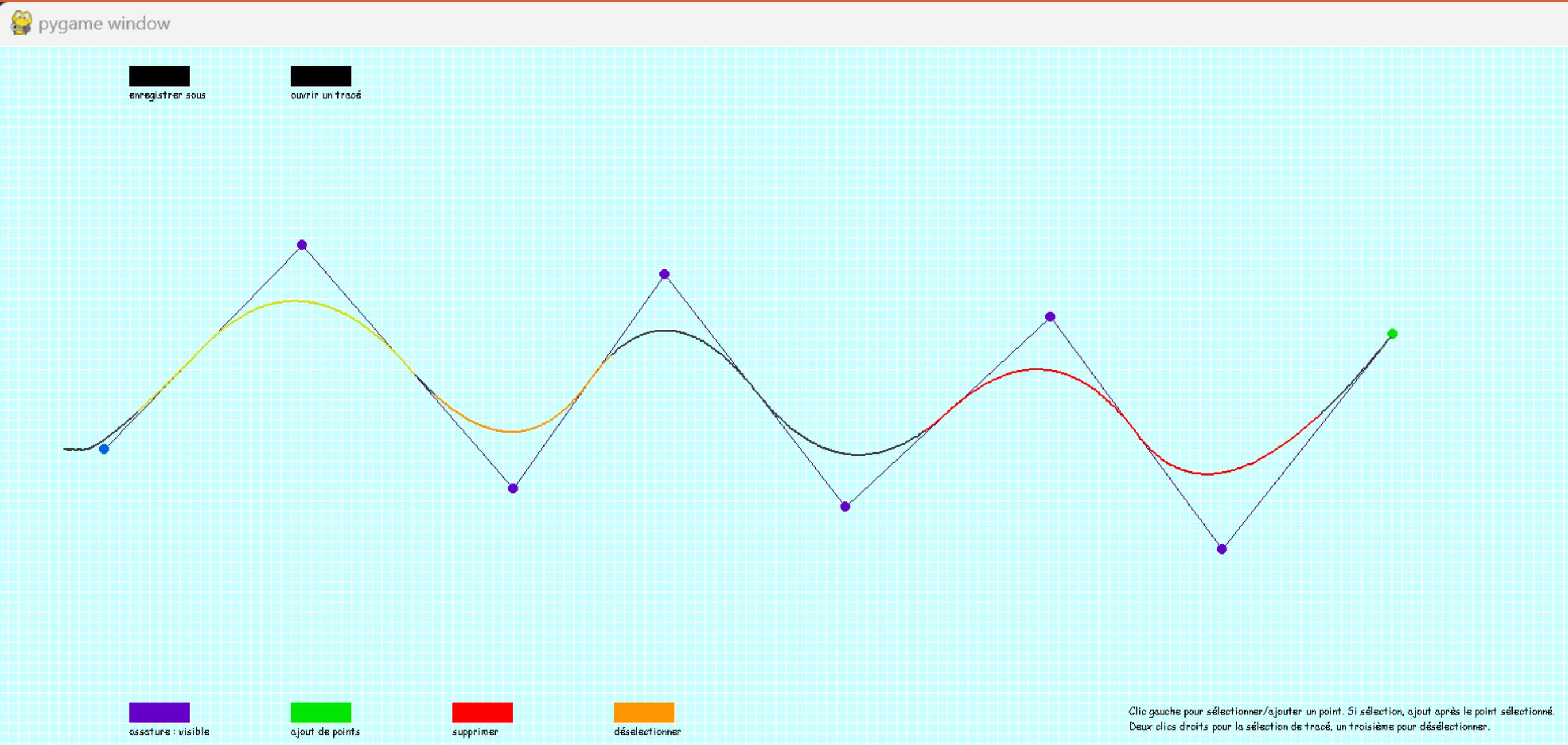
# Simulation finale

# Différents types de rails et une plateforme de départ

- Le rail est toujours tracé par l'utilisateur.
- Une plateforme de pente nulle apparaît également au début du rail pour le départ du wagon

- L'utilisateur peut choisir des parties de rails qui freinent, accélèrent ou entraînent le wagon à vitesse constante.

Clément-Anouchka



simulation finale

# Acquisition de données physiques sur le wagon

- On peut obtenir la vitesse et l'accélération du wagon au cours du temps.
- L'énergie mécanique, cinétique et potentielle du wagon est calculée à chaque instant.
- On calcule également l'énergie électrique consommée par le wagon pour un circuit donné.

# Conclusion

**Simulation physique interactive  
qui permet de visualiser les  
paramètres physique d'un objet**

