# Guillaume Lay

Mémoire

# En quoi un choix de langage informatique fige-t ’il un projet dans le temps ?

2020-2021

Table des matières

[Guillaume Lay 1](#_Toc48817832)

[En quoi l’apparition de langages de programmation est-elle nécessaire ? 1](#_Toc48817833)

[Remerciements 2](#_Toc48817834)

[Intro, comment parler à une machine 5](#_Toc48817835)

[L’origine du bit 5](#_Toc48817836)

[Notion de valeur 5](#_Toc48817837)

[Les couches applicatives 5](#_Toc48817838)

[De plus en plus accessible 5](#_Toc48817839)

[Partie 1 : Qu’est-ce qu’un langage de prog 6](#_Toc48817840)

[Utilité 6](#_Toc48817841)

[Plateforme d’utilisation/Ressources 6](#_Toc48817842)

[Mode d’exécution 6](#_Toc48817843)

[Syntaxe 6](#_Toc48817844)

[POO 6](#_Toc48817845)

[Librairies 6](#_Toc48817846)

[Partie 2 : L’évolution de la technologie et des langages de prog 7](#_Toc48817847)

[Notions d’OS 7](#_Toc48817848)

[Evolution des supports 7](#_Toc48817849)

[Simplification des langages 7](#_Toc48817850)

[Partie 3 : Comment créer un langage de prog 8](#_Toc48817851)

[Compilateur 8](#_Toc48817852)

[Esolangs 8](#_Toc48817853)

[Partie 4 : Pourquoi le besoin de nouveaux langages 9](#_Toc48817854)

[Langages qui meurent 9](#_Toc48817855)

[Nouvelles performances 9](#_Toc48817856)

[Partie 5 : L’impact d’un choix de technologie 10](#_Toc48817857)

[Apprentissage 10](#_Toc48817858)

[Limitations 10](#_Toc48817859)

[Résistances 10](#_Toc48817860)

[Partie 6 : Un esperanto informatique 11](#_Toc48817861)

[Caractéristiques 11](#_Toc48817862)

[Implications 11](#_Toc48817863)

[Partie 7 : Les langages du futur 12](#_Toc48817864)

[Vers du no code ? 12](#_Toc48817865)

[Accessible à tous ? 12](#_Toc48817866)

[Ordinateur quantique ? 12](#_Toc48817867)

[Conclusion 13](#_Toc48817868)

[Bibliographie 14](#_Toc48817869)

[Livre 1 14](#_Toc48817870)

[Livre 2 14](#_Toc48817871)

[Livre 3 14](#_Toc48817872)

[Déroulé méthodologique 15](#_Toc48817873)

[Présentation du thème : 15](#_Toc48817874)

[Liste des personnes rencontrées : 15](#_Toc48817875)

[Calendrier M2 15](#_Toc48817876)

## Remerciements

Je tiens à remercier ILTR bidule machin.

## Résumé

## Abstract

# Introduction : qu’est-ce qu’un langage informatique ?

Logique, interprétation par l’ordinateur etc…

### L’origine du bit

Vrai ou faux ? Voici la question qui dirige le monde moderne. A l’heure ou l’informatique s’est installée dans notre quotidien, il n’est plus l’heure des « peut-être ».

La langue maternelle de tout matériel informatique est le binaire. C’est l’interprétation directe de ce qui est physiquement écrit dans la machine. Que ce soit un trou dans une carte perforée ou la charge d’un électron, le fondement de l’information est un état physique à deux issues. Cela forme le dialecte le plus basique formé de 1 et de 0.

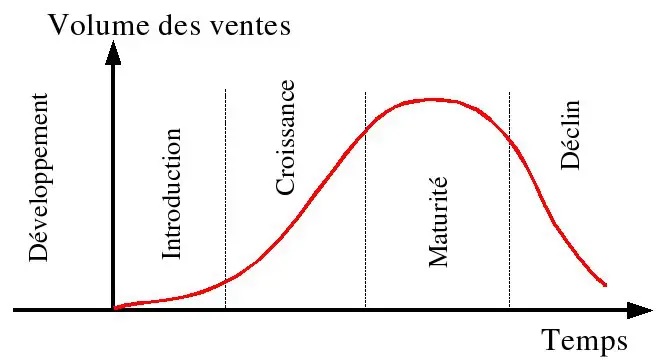
Jusqu’au temps où l’humain fusionnera avec la technologie, nous ne sommes actuellement pas en mesure de discuter directement avec un ordinateur. Nous utilisons toute sortes de langages afin de communiquer avec ces amas de silicium et de circuits électroniques. JavaScript, PHP, C++… il y en a des centaines et de nouveaux apparaissent tous les ans. Pourquoi il y en a-t-il autant ? Pourquoi sont-ils si différents ? Et surtout, pourquoi en invente-t-on de nouveaux ? C’est à ces interrogations que nous allons tenter de porter des réponses.

# Partie 1 : L’informatique et le temps

## Introduction

Comme tout produit, chaque langage informatique suit un cycle de vie composé de cinq phases distinctes. Chacune de ces périodes possède ses propres questionnements et enjeux pour atteindre les objectifs de ventes ou de taux d’utilisation.

Ces cinq étapes sont les suivantes :

* Développement : Toute la partie conception et création du produit, l’établissement de ses règles et propriétés.
* Introduction : Le moment où le produit est révélé aux utilisateurs.
* Croissance : La période pendant laquelle le produit gagne en notoriété et se voit évoluer rapidement.
* Maturité : Lorsque le produit est installé et profite d’un taux d’utilisation maximal.
* Déclin : Le produit cesse d’être utilisé et tombe en désuétude.

Dans cette partie, nous détaillerons ces différentes étapes du cycle de vie d’un produit en ce concentrant sur les points les plus importants afin d’observer l’impact du temps sur un produit informatique. Pour cela, nous nous aiderons d’un cas concret, celui du projet Leek Wars.

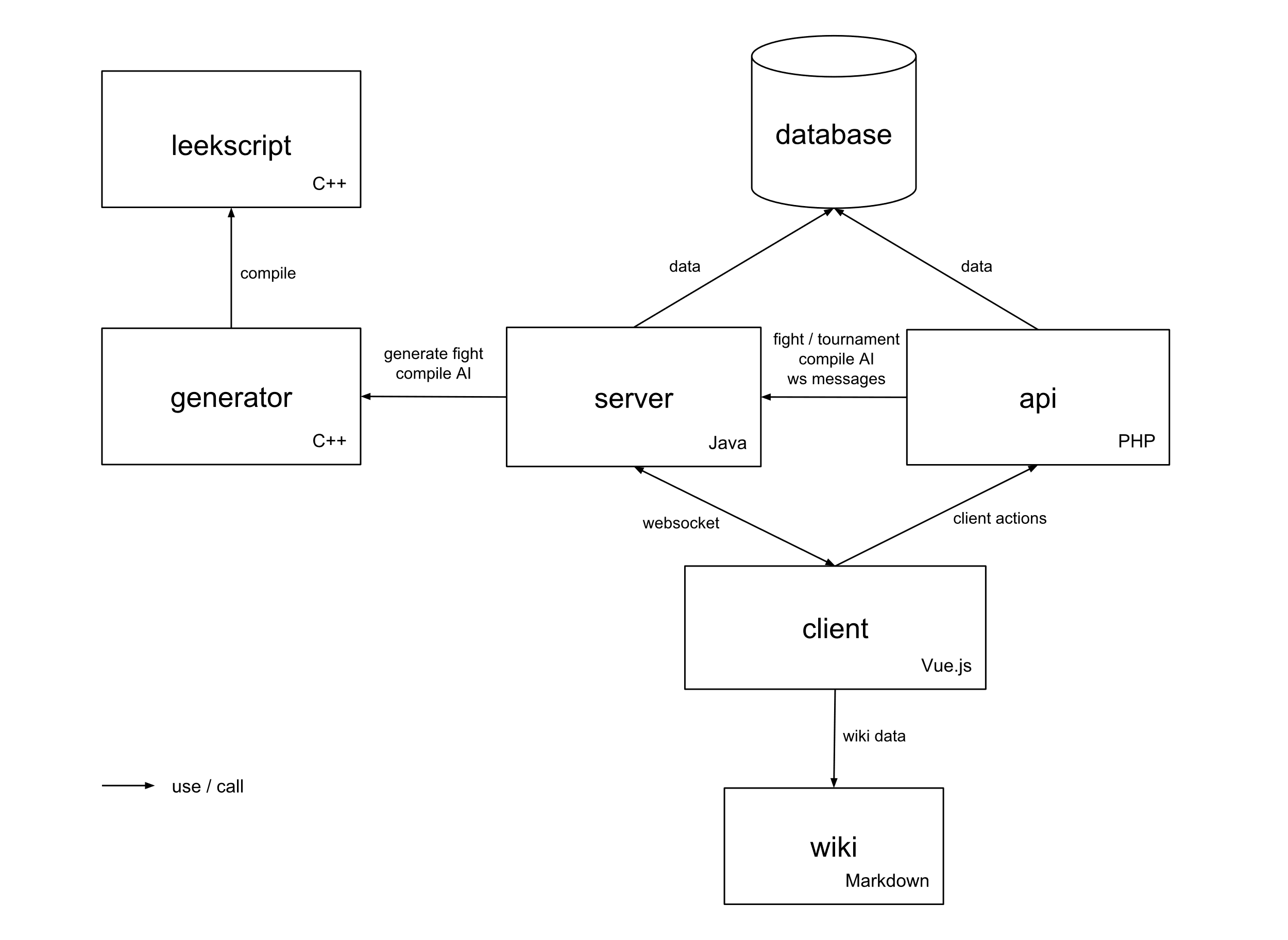
## Leek Wars

### Résumé

Leek Wars est un jeu vidéo français créé en 2013 par des étudiants en informatique. L’objectif de ce projet est d’initier les joueurs à la conception d’intelligences artificielles et la programmation en général. Le principe est que chaque joueur dispose d’un personnage à l’effigie d’un poireau et doit affronter les personnages adverses. Pour cela, les actions que le poireau devra effectuer lors de l’affrontement devront être définie au préalable par le biais d’un script écrit par le joueur.

Sachant que les combats se déroulent au format tour par tour, à la manière d’un jeu d’échecs, le comportement du poireau doit prévoir toutes les situations afin d’espérer vaincre son ennemi. Toute une panoplie d’armes et de sortilèges sont à la disposition des combattants afin qu’ils usent de stratégie et de fins calculs pour que leur poireau ait le plus de chances de se sortir de toutes les situations.

Même si le projet est réservé à un public très ciblé, à savoir les curieux au sujet de l’informatique et les développeurs en quête d’expérimentation, le jeu a su gagner en notoriété au fil des ans et la communauté ne cesse de s’agrandir. La barre des cinquante mille inscrits a été dépassée et l’application est en constante évolution grâce à des mises à jour régulières apportant équilibrage et nouveau contenu.



Le projet pourrait être comparé à un Framework entier puisqu’il comprend à lui seul son langage, son compilateur ainsi que son propre IDE.

Questions à Pilow :

Que fais-tu dans la vie ?

Comment est venue l’idée de Leek Wars ?

Comment ont été choisies les technos du projet ?

Comment a été créé le LeekScript ? Choix de syntaxe, fonctionnement

Quelles sont les limitations du LeekScript V1 ?

Qu’est-ce qui a motivé la création du LSV2 ? Quelles difficultés rencontrées ?

A partir de quand décide t’on de modifier existant ou de repartir de zero ?

Combien de temps estime tu la durée de vie du projet ?

## Sous-partie 1 : La création d’un langage de programmation

### Le besoin

Avant de lancer la création d’un nouveau langage de programmation, il est nécessaire de connaître ses motivations. Il doit combler un manque, apporter quelque chose de nouveau.

Créer un compilateur de langage,

Esolangs, cette mine d’or

## Sous-partie 2 : Comment faire évoluer son produit

## Sous-partie 3 : Comment anticiper le déclin

# Partie 2 : Comment choisir son langage

# Partie 3 : L’impact d’un choix de technologie

Cas d’étude GEODP V1 et V2

Gérer la transition

## GEODP

Description du projet

Partie gestion de projet

Formation de l’équipe

## Apprentissage

## Limitations

## Résistances

# Conclusion

# Bibliographie

## Livre 1

## Livre 2

## Livre 3

Déroulé méthodologique :

### Présentation du thème :

Durant tout mon parcours informatique, dès que je parlais de langages de programmation à des non-initiés, j’ai toujours eu droit à cette interrogation : « Mais pourquoi y en a-t-il autant ? ». Même si j’ai une très vague idée de cette raison, j’ai trouvé intéressant de répondre une bonne fois pour toutes à cette question. C’est un sujet qui me passionne et en l’abordant, il me permettra de faire un bilan complet de mes compétences acquises jusqu’à présent.

Ma thématique est assez vague ce qui me permettra d’aborder un large panel de sujets différents.

Liste des personnes rencontrées : 0

### Calendrier M2