

Qui est-ce ?

Projet de programmation

Groupe AB
Martinet Hugo, Xihao Wang, Ziyu Gao
<https://github.com/m-hugo/GuessWho>
L2 informatique
Faculté des Sciences
Université de Montpellier.

4 avril 2022



Résumé

Description très succincte du problème et des différentes étapes de réalisation

1 Technologies utilisées et organisation

Une page.

1.1 Choix du langage

- Le langage rust est utilisé pour l'intégralité du projet
- La bibliothèque graphique fltk-rs est utilisée pour l'affichage de la fenêtre et des widgets, la fonctionnalité de gestion des événements par message incluse avec les widgets est très pratique

La bibliothèque graphique egui a été fortement considérée et un prototype a été créé mais le concept d'"immediate mode" n'était pas facile à utiliser quand on fait une application qui doit initialiser beaucoup de données pendant l'exécution, garder beaucoup d'états en mémoire et avoir des widgets qui en influencent beaucoup d'autres

- Le framework rocket est utilisé pour avoir un serveur similaire à node.js (mais avec la vitesse de rust!)
- La bibliothèque reqwest est utilisée côté client pour accéder au serveur

- La bibliothèque `rand` est utilisée pour tirer un personnage aléatoire, le tirage de nombres aléatoires n'étant pas présent dans la bibliothèque standard de `rust`
- La bibliothèque `serde` est utilisée pour importer et exporter le `json` (planches et sauvegarde)

1.2 Organisation du travail

1. Répartition du travail au sein du groupe

J'ai commencé à faire le prototype tout seul avant de trouver un groupe puis j'ai travaillé avec Xihao Wang, Ziyu Gao à partir de la première séance, nous avons fait beaucoup de code ensemble mais ils ont aussi fait du code tout les 2, publié sous `xihao-wang`, et j'ai aussi fait pas mal de code moi même, publié comme le travail de groupe sous `m-hugo`

J'ai fait l'architecture des programmes ainsi que le choix des bibliothèques, les deux parties les plus compliquées quand on fait un projet en `rust` car une bonne architecture nous permet de rendre le programme beaucoup plus interactif en évitant les problèmes de "borrowing", d'"ownership" et de durée de vie des variables. Les bibliothèques `rust` sont très nombreuses et la plupart ne marchent pas ou manquent de fonctionnalités

2. Rythme de travail, mode de fonctionnement ...

A la première séance je leur ai appris les bases du `rust`, nous avons ensuite travaillé ensemble 2-3 heures par semaine le lundi et nous finissions certaines tâches de notre côté pendant la semaine

2 Étape 1 : permettre à l'utilisateur de jouer

3 pages

- Fonctionnalités de l'application : interactions possibles de l'utilisateur.
- Format du fichier `JSON`, contraintes éventuelles.

```
{
  "attrs": {
    "Nom": ["nom1", "nom2"],
    "attribut1": ["possibilité1", "possibilité2"],
    "attribut2": ["possibilité1"]
  },
  "liste": [
    {
      "Nom": "nom1",
      "attribut1": "possibilité1",
      "attribut2": "possibilité1",
      "image": "./personnages/nom1.png"
    },
    {
      "Nom": "nom2",
      "attribut1": "possibilité2",
      "attribut2": "possibilité1",
      "image": "./personnages/nom2.png"
    },
    (22 autres "{}")
  ]
}
```

"attrs" contient tout les attributs et toutes les possibilités pour chaque attribut "liste" contient 24 blocs, chacun contenant tous les attributs definis dans "attrs" suivi de "image" avec le chemin de l'image depuis ce dossier ou un chemin absolu

- Description de la forme des requêtes traitées.
- Description des structures de données, classes, variables utilisées.
- Description du traitement effectué lors d'une requête.

3 Étape 2 : aider à la saisie des personnages

2 pages

- Description du problème : format des données et du résultat.
- Scénario des interactions avec l'utilisateur.
Nous avons prévu de permettre aux utilisateurs de valider les attributs/valeurs avec la touche entrée pour pouvoir en saisir plusieurs à la suite sans utiliser la souris (il est possible de savoir quelle boîte à le focus) mais nous n'avons pas réussi

4 Étape 3 : XXXX

5 pages

Description précise de l'extension réalisée - choix éventuels

Description de la résolution (protocole, algorithmes, ... selon l'extension choisie) Nous utilisons la methode node.js, le seueur execute differentes fonctions selon le format de l'url utilisée pour y acceder et renvoie une chaine de caracteres

5 Bilan et Conclusions

1 page

Ce qui a fonctionné, ce qui n'a pas fonctionné, problèmes rencontrés, ...