

Présentation du projet :

- Pour notre projet nous nous sommes inspirés du jeu de cartes des sept familles scientifiques et en même temps du jeu de bataille qui se joue avec un jeu de cartes classique. Nous avons donc eu l'idée de créer un jeu de bataille avec les cartes des sept familles ! Afin d'approprier une force à chaque carte nous avons utilisé le chiffre en binaire qui est indiqué sur chaque carte en haut à droite. Après la création du jeu de bataille classique en python, le groupe a décidé de créer son propre jeu de carte. Nous avons appelé notre jeu « le collectionneur de scientifiques ». Pour gagner, un joueur doit collectionner tous les scientifiques d'une même famille ou tous les scientifiques de la même force.
- Les règles du jeu :
 - Les joueurs gardent dix cartes dans leur main tout au long de la partie, le reste des cartes est mis dans une pioche
 - Le gagnant de chaque confrontation choisie entre deux options : 1- Il prend les deux cartes et donne une carte de son paquet à son adversaire/ 2- Il met les cartes jouées dans une « poubelle », prend deux cartes de la pioche et donne une carte de son choix à son adversaire.
 - Si lors de la confrontation il y'a une égalité, les deux cartes sont mises dans la « poubelle » et chaque joueur prend une carte de la pioche.
 - Lorsque la pioche est finie, la « poubelle » est mélangée et devient votre nouvelle pioche.
 - Le premier qui collectionne 7 cartes d'une même force ou toutes les cartes d'une même famille gagne la partie !

Validation du projet :

- Après avoir testé le jeu à plusieurs reprises, les problèmes (la majorité des problèmes étaient visuels (user interface avec tkinter)) ont été détectés et réglés. Notre jeu est maintenant optimal et les joueurs peuvent jouer sans aucun problème !

Présentation des étapes du projet :

- 1- Création du site HTML qui présente les scientifiques.
- 2- Réfléchir en groupe sur un jeu original à créer (règles du jeu...)
- 3- Documents visuels (photos, vidéos...)
- 4- Création du code de la bataille classique en python
- 5- Création du code de la deuxième version du jeu
- 6- Validations du projet
- 7- Réfléchir sur des pistes futures et une amélioration du projet
- 8- Création du document descriptif + présentation du jeu en 500 caractères.
- 9- Envoi du projet

Ouverture et idées d'améliorations :

- On prévoit de créer une intelligence artificielle qui sera entraînée pour jouer contre un joueur à notre jeu, et d'améliorer les possibilités de stratégies à faire pour gagner. Pour cela,

il nous faut beaucoup de données, de parties jouées, permettant de créer notre propre model. De plus, notre future intelligence artificielle devrait avoir un niveau de difficulté disponible (à choisir par le joueur).

Aussi, on prévoit rajouter notre jeu en fichier exécutable sur le site web accompagnant notre projet, ce qui permet de centraliser tout le projet sur le site web.

Enfin, il serait idéal d'améliorer l'interface graphique avec tkinter pour la rendre plus esthétique et plus facile à manipuler.

Organisation du travail :

Présentation des élèves :

- Mathieu Pierre RIZKALLAH : Elève de spécialité NSI au GLFL en 1^{er}
- Zouhair EL KOUSH : Elève de spécialité NSI au GLFL en 1^{er}
- Alexandre Mathieu HINDI : Elève de spécialité NSI au GLFL en 1^{er}
- Cyril ABDELNOUR : Elève de spécialité NSI au GLFL en 1^{er}
- Wissam DAOUK : Elève de spécialité NSI au GLFL en 1^{er}

Répartitions du travail :

Mathieu :

- Recherche du sur les personnages pour alimenter les sites web
- Création des prototypes
- Création des fonctions python
- Création des codes tkinter permettant l'interface graphique
- Test du jeu
- Résolution des erreurs obtenues
- Code en HTML CSS
- Rédaction de L'ouverture et de la documentation
- Elaboration des règles du jeu de carte

Zouhair :

- Photoshop des images des cartes du jeu
- Montage de la vidéo
- Création du prototype du jeu de bataille classique
- Création des codes tkinter pour l'interface graphique
- Recherche des personnages pour alimenter les sites web
- Code du HTML et du CSS du site

Wissam : - recherche du sur les personnages pour alimenter les sites web

- Création des prototypes
- Création des codes tkinter permettant l'interface graphique
- Création des fonctions python
- Test du jeu
- Résolutions des erreurs remarquées+ optimisation du projet
- Code en HTML CSS

Alexandre : - recherche du sur les personnages pour alimenter les sites web

- Proposition de prototypes
- Choix des couleurs et des images
- Test du jeu+ résolution de problèmes
- Désigne de la structure de la page web
- Création du document descriptif
- Création de la présentation en 500 caractère.

Cyril :

- recherche du sur le personnage pour alimenter les sites web
- Proposition de prototypes
- Choix des couleurs et des images
- Test du jeu+ résolution des problèmes
- Rôle de communication dans le groupe

Tout au long du projet, on s'est réunis plusieurs fois dans l'établissement durant les pauses méridiennes et en dehors de l'établissement pendant les vacances et les week-end. Ce qui nous a permis de garder une bonne communication au sein de l'équipe et d'avancer proprement et efficacement.

On a aussi utilisé divers outils : GitHub, Google drive, Photoshop, Visual studio code... Ces outils nous ont été nécessaires pour l'avancement du jeu.

Documentation :

Pour jouer le jeu de la bataille classique, il suffit de lancer le code bataille_classique.py et de suivre les étapes (choisir le mode de jeu d'abord : contre l'ordinateur ou contre un joueur, puis appuyer sur commencer pour initialiser le jeu).

Pour jouer le jeu qu'on a créé (variante de la bataille et des 7 familles), il suffit de lancer le code variante.py et de suivre les étapes demandées (il suffit de choisir une carte par joueur et d'appuyer sur Jouer pour commencer la bataille, mais à vous de choisir la meilleure stratégie pour tenter de gagner 😊)

Les codes sont basés sur la librairie Tkinter qui nous sert d'interface graphique. Ainsi tout notre projet est structuré sous cette forme : d'un côté les codes Tkinter qui permettent la mise ne place de l'interface d'utilisateur, et d'un autre coté les fonctions pythons qui permettent l'exécution des fonctionnalités. On utilise aussi les libraires random et PIL (pour les images)

Les langages utilisés : Python, HTML, CSS et JS

On a choisit de tout centralisé sur GitHub pour une meilleur gestion du code et de l'organisation des fichier entre le groupe.

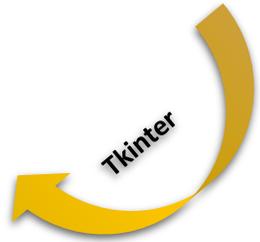
```

from tkinter import *
from random import *
from PIL import Image, ImageTk

root = Tk()
root.title('Le collectionneur de cerveaux informatiques, le jeu des cartes des sciences numériques')
icon = PhotoImage(file = "images/icon.png")
root.iconphoto(False, icon)
root.attributes('-fullscreen', True)
root.configure(background="#2D2727")
icon = PhotoImage(file = "images/icon.png")
root.iconphoto(False, icon)
root.resizable(0,0)
mainGrid = Frame(root)
mainGrid.configure(background="#2D2727")
username1 = Label(mainGrid, text= "Joueur 1", font=("Helvetica", 16))
username2 = Label(mainGrid, text= "Joueur 2", font=("Helvetica", 16))

mainGrid.columnconfigure(0, weight=1)
mainGrid.columnconfigure(1, weight=1)
mainGrid.columnconfigure(2, weight=1)
username1.grid(row=0, column=0, pady= 20)
username2.grid(row=0, column=2, pady= 20)
mainGrid.pack(fill="x")

```



```

> def graveyardSize(card): ...
> def resize_cards(card): ...
> def resizeGrid(card): ...
> def init(): ...
> def Commencer(): ...
> def distributeDeck() : ...
> def deckDisplay (): ...
> def GraveyardCreate(): ...
> def button_press1(event): ...
> def button_press2(event): ...
> def Quit(): ...
> def message(message): ...
> def JouerRemove(): ...
> def createcard(carte): ...
> def GiveOther_1_2 (): ...
> def GiveOther_2_1 (): ...

```



```

> def GiveOther2_1 (): ...
> def tkCard1(event): ...
> def tkCard2(event): ...
> def bruler_1(event): ...
> def bruler_2(event): ...
> def egalite(): ...
> def Jouer(): ...

```



```

holderLabel1 = Label(mainGrid, text = '.')
holderLabel2 = Label(mainGrid, text = '.')
holderLabel1.grid(row= 1, column= 0)
holderLabel2.grid(row= 1, column= 2)

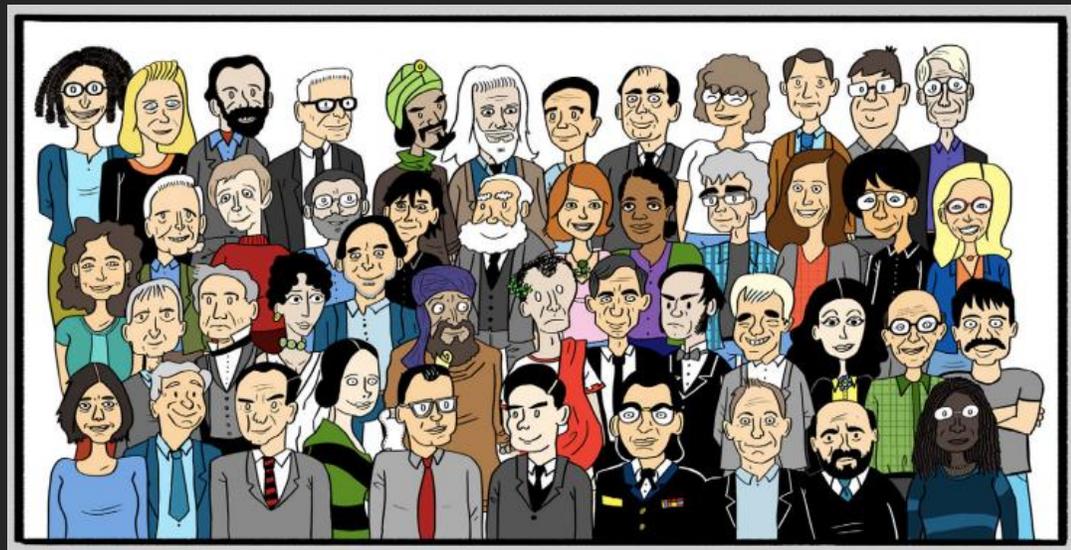
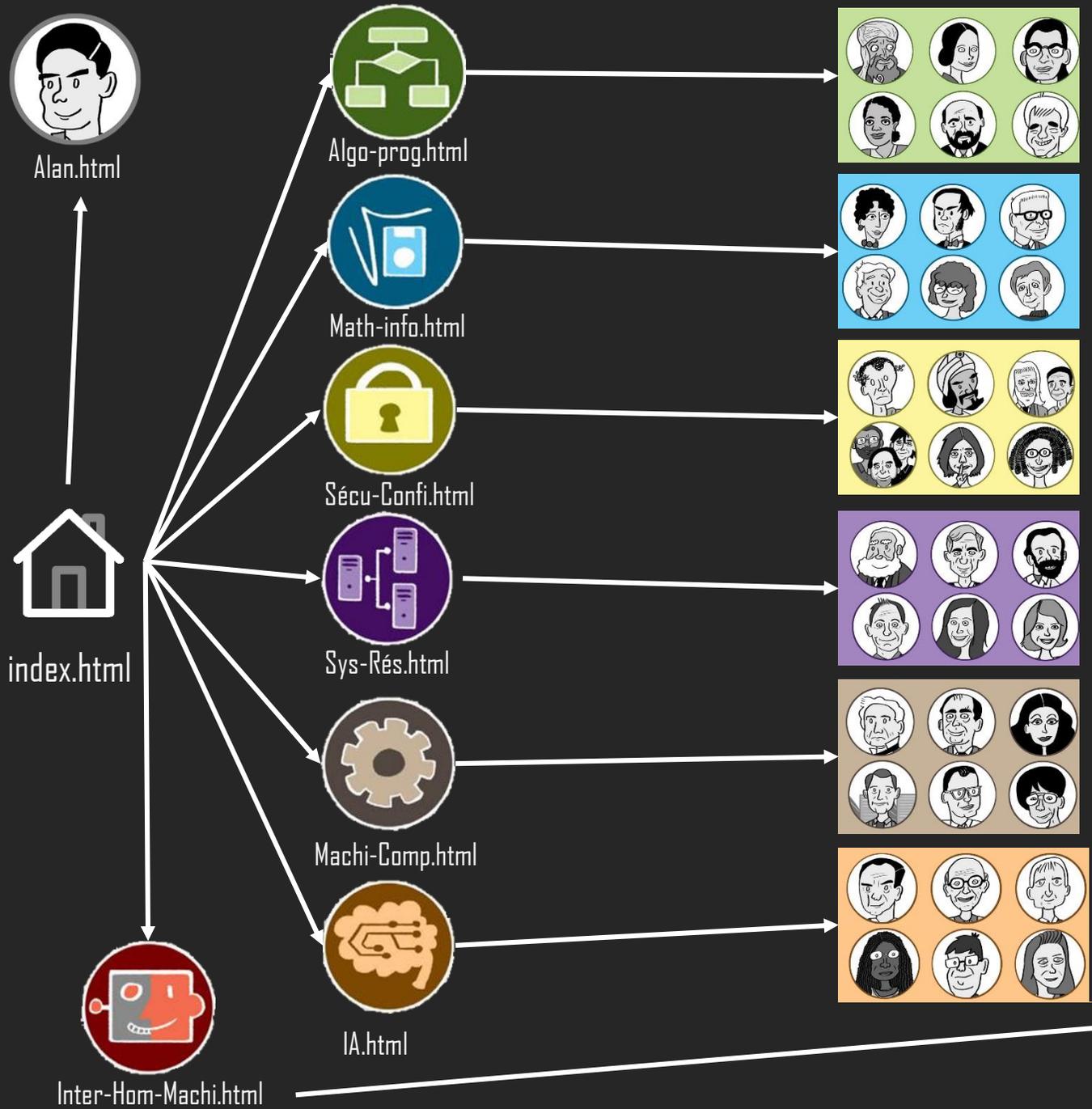
startButton = Button(mainGrid, text="Commencer", command=Commencer, width=30, font=("Helvetica", 18), borderwidth=0)
startButton.grid(row= 1, column= 1)
quitButton = Button(mainGrid, text="Sortir", command=Quit, width=20, font=("Helvetica", 18), borderwidth=0)
quitButton.grid(row= 0, column= 1)
listCards = [ ...
dictionnaryCards = { ...
def restart (): ...

def conditionCheck(): ...

init()
root.mainloop()

```





FICHIERS DE SOURCE



REPARTITION DU TRAVAIL: roles A

Zouhair Elkoush:

- Mettre en commun les différents pages pour les familles
- Prototype page d'accueil
- Codage des pages final et du pop up
- Charte graphique
- Cree un group WhatsApp avec les autres rôles A des 2 classes et un autre pour la famille
- Regrouper les informations données par les groupe
- Rechercher Al-Khwarizmi

Mathieu Rizkallah:

- Création de prototypes
- Prototype de la page de famille
- Choix des couleurs
- Coordination au sein de son group(avec roles B)
- Mise en place du css et html de base
- Intro de sys - res

Alexandre Hindi:

- Cree et organiser le drive
- Coordination au sein de son group(avec roles B)
- Regrouper les informations données par les groupe
- Introduction général du site (page d accueil)
- Introduction de machine et composant
- Recherche sur le personnage Hiroshi

Angéline Samaha:

- Coordination au sein de son group(avec roles B)
- Aide dans le design et couleur
- Recherche des codes
- Recherche de Joelle Coutaz
- Introduction inter homme-machine

Carine Abbas:

- Organization du design
- Recherche des nouvelles syntaxes
- Préparer les images de chaque personnages et familles
- Coordination au sein de son group(avec roles B)

Zeidan Oueidat:

- Coordination au sein de son group(avec roles B)
- Recherche de nouveau code
- Tests de la plateforme finale

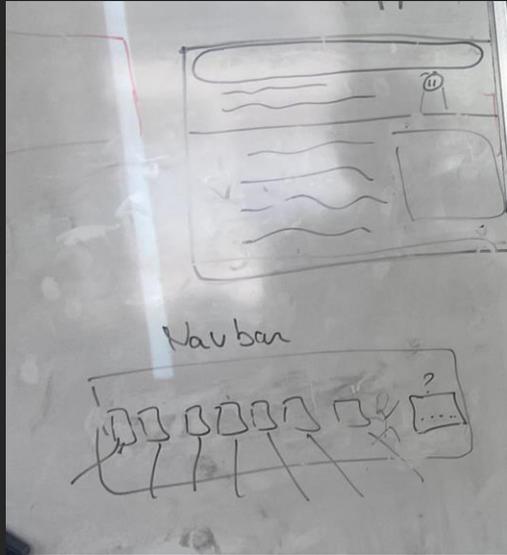
Lynn Abla:

- Carte mentale
- Nomenclature des fichiers
- Coordination au sein de son group(avec roles B)

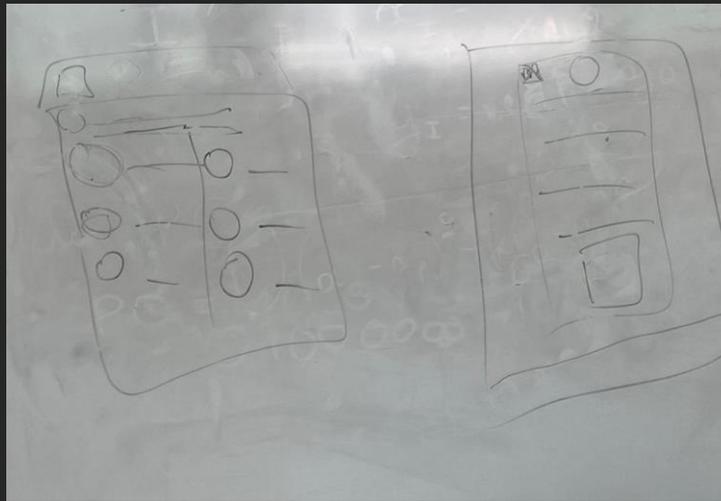
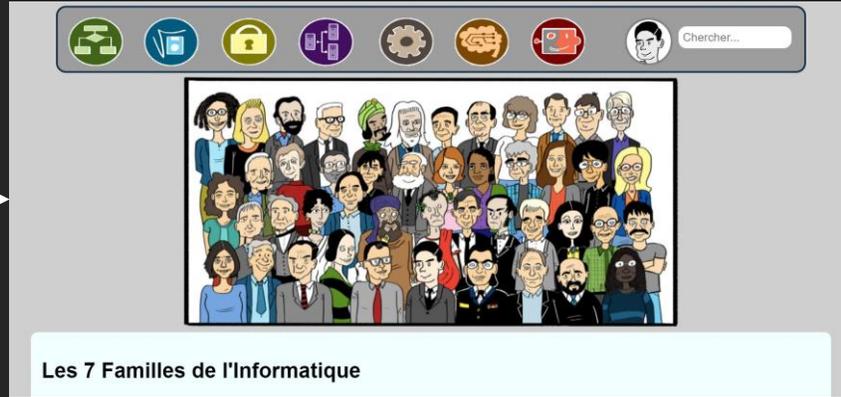
Issam Karam:

- Page Alan Turing
- Présentation du personnage
- Recherche de nouveau code

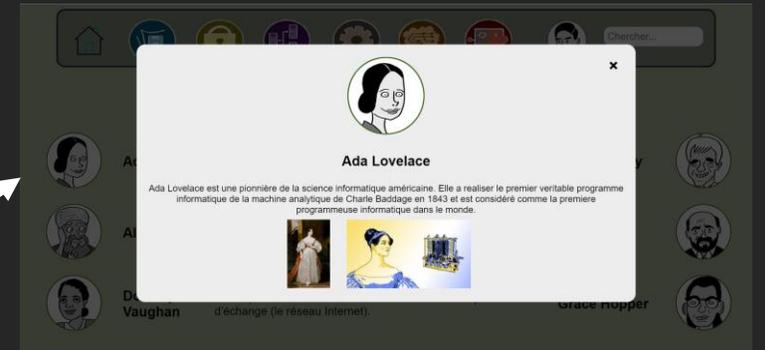
Idées desiner du groupe I de NSI:



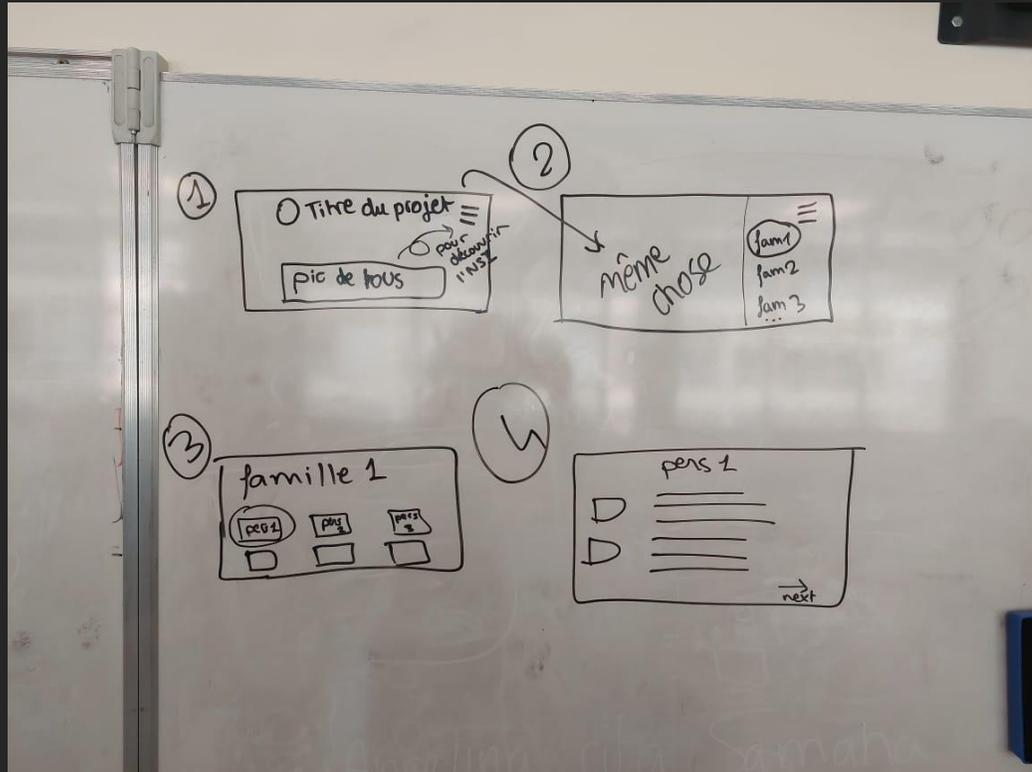
Page d'accueil



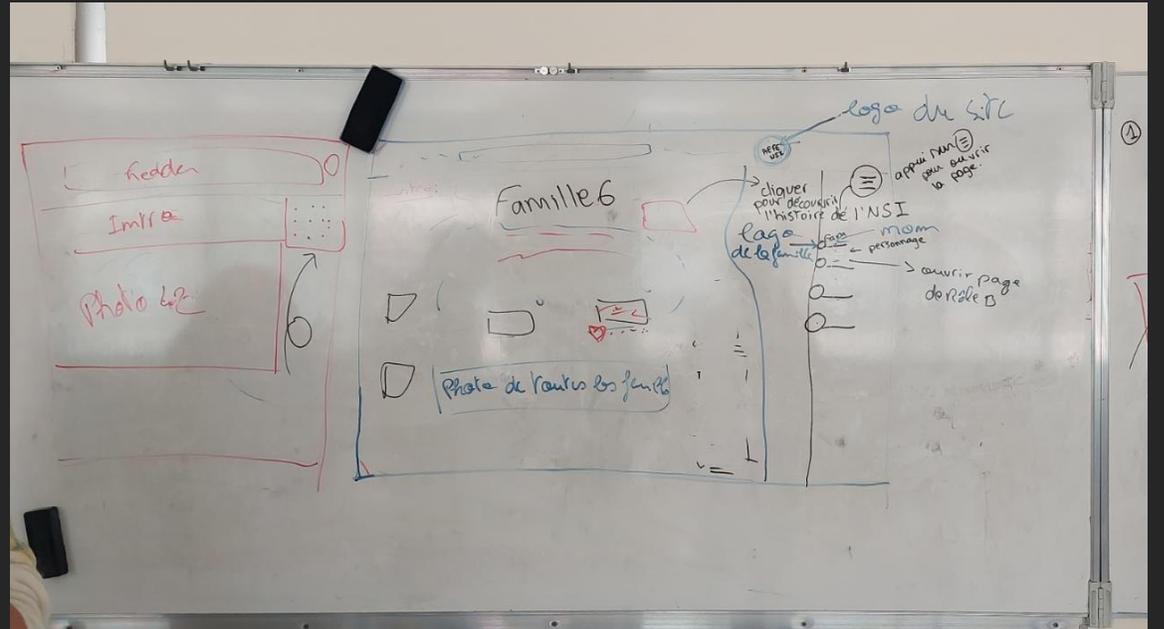
Page de familles



Idées desiner du groupe 2 de NSI:



Différentes pages mit en commun



Design des pages