



RAPPORT DES PROJETS :

ASTRE-IPS

&

CMC

Document rédigé par :
Radhwen SAIDI



I- Projet ASTRE - IPS

Introduction

Le projet ASTRE-IPS a pour objectif de faire des prédictions de l'orientation des étudiants informatique de l'ensim vers les deux filières de l'école ASTRE ou IPS. La prédiction se base sur les résultats des réponses des étudiants de 3A sur le formulaire proposé par les étudiants de 5A IPS d'une part, et sur les hypothèses fixés par notre groupe de projet d'autre part.

Chaque étudiant reçoit un score qui correspond à nombre de points accumulés par le calcul du produit du poids de l'hypothèse et le nombre de points qu'elle offre. (positif si IPS, négatif si ASTRE).

L'interface graphique de ce projet est sous la forme d'un site web.

L'utilisateur peut interagir avec l'outil en changeant les poids des hypothèses, et ce en jouant avec les sélectionneurs qui se trouvent dans la page d'accueil.

Ce projet est le résultat de plusieurs étapes enchainées d'une manière logique afin de séparer les couches "separation of concern" et représenter les données de la manière la plus efficace possible.

NB: une vidéo de démo est jointe avec ce document

I-1- Extraction et nettoyage des données:

Le dataSet de ce projet est l'ensemble des réponses des étudiants sur le questionnaire envoyé par les étudiants de 5A. Ces réponses, brutes, sont stockées dans un fichier csv. Ce qui implique que la première étape consiste à extraire les données, effectuer le nettoyage des champs qui représentent des questions non pertinentes pour le calcul, remplacer les textes des questions par des valeurs plus lisibles et exploitables par le compilateur et finalement générer un fichier JSON qui représente toutes ses données pour la prochaine partie de traitement. Toutes ces étapes ont été réalisés en utilisant python comme étant le langage ayant le plus de bibliothèque pour manipuler les fichiers csv.

	Horodateur	Quel est votre numéro étudiant ?	Êtes-vous de bonne humeur ?	Qu'avez-vous fait avant l'ENSIM ?	Suivez-vous les actualités technologiques ?	Avez-vous des passe-temps artistiques ?	Êtes-vous créatif ?	Êtes-vous ?
0	2021/09/21 11:50:37 AM UTC+2	20190448	3	Prépa intégrée	2	3	3	BDLC;Enigma;Ens
1	2021/09/21 11:51:39 AM UTC+2	e190568	3	Prépa intégrée	2	4	3	
2	2021/09/21 11:52:05 AM	21005534	4	Prépa	3	4	4	



ENSIM

École d'ingénieurs
Le Mans Université

```
[ ] ipsastre=ipsastre.drop(['Êtes-vous de bonne humeur ?', 'Horodateur'], axis=1)

[ ] ipsastre=ipsastre.rename(columns={"Quel est votre numéro étudiant ?": "numEtu", "Qu'avez-vous fait avant d'arriver à ENSIM ?": "beforeEnsim"}, inplace=True)

[ ] ipsastre=ipsastre.replace(to_replace = "Prépa intégrée",
                             value = "Prepa integree")

[ ] ipsastre=ipsastre.replace(to_replace = "Prépa classique",
                             value = "Prepa classique")

[ ] ipsastre=ipsastre.replace(to_replace = "Prépa ATS",
                             value = "Prepa ATS")
```



```
data = ipsastre.to_json('./export.json', orient='records')
print(data)
```

```
[ ]
{
  "numEtu": "20190448",
  "beforeEnsim": "Prepa integree",
  "actTech": 2,
  "tmpsArt": 3,
  "creatif": 3,
  "assosEtu": "BDLC;Enigma;Ensimien;Kfet;Troublions du plateau",
  "soirEnsim": 3,
  "sitClasse": "Plutot milieu",
  "netflixAmazon": "Oui",
  "animes": 2,
  "systExpl": "Windows",
  "monterPc": "Jamais",
  "nbEcran": "Non",
  "cpuGraph": "Performances",
  "rienBase": "A partir d'une base",
  "equipe": 2,
  "dirige": "Diriger",
  "colocation": "Non",
  "keyMouse": "Clavier"
},
{
  "numEtu": "190568",
  "beforeEnsim": "Prepa integree",
  "actTech": 2,
  "tmpsArt": 4,
  "creatif": 3,
  "assosEtu": "Aucune",
  "soirEnsim": 2,
  "sitClasse": "Plut\u00f4t derri\u00e8re",
  "netflixAmazon": "Oui",
  "animes": 4,
  "systExpl": "Windows",
  "monterPc": "Jamais",
  "nbEcran": "Oui",
```



I-2- Élaboration des hypothèses et calcul des points:

Cette partie a été développée en utilisant le framework .NET C#. Elle prend en entrée le fichier JSON généré par le script python qui contient les réponses de chaque étudiant aux questions. Le script C# consiste à spécifier toutes les hypothèses et leurs poids d'une part, et le calcul du nombre de points que chaque étudiant a eu pour chaque réponses. L'étudiant reçoit par la suite des points bonus qui sont le résultat des hypothèses composées dans l'algorithme.

```
[
  {
    "numEtu": "20190448",
    "Q17": "0",
    "Q23": "2",
    "Q32": "2",
    "Q42": "1.5",
    "Q51": "-5",
    "Q52": "3",
    "Q61": "2.5",
    "Q72": "0",
    "Q81": "3",
    "Q91": "3",
    "Q102": "3",
    "Q112": "-3",
    "Q121": "-4",
    "Q132": "-3",
    "Q142": "-4",
    "Q151": "3",
    "Q162": "-1",
    "Q171": "-5",
    "Bonus": -6
  },
  {
    "numEtu": "190568",
    "Q17": "0",
    "Q23": "2",
    "Q31": "4",
    "Q42": "1.5",
    "Q53": "1",
    "Q62": "-2.5",
    "Q81": "3",
    "Q91": "3",
    "Q102": "3",
    "Q111": "3",
    "Q122": "-5"
```

```
var hypotheses = new Map([
  ["Q11", 4],
  ["Q12", 4],
  ["Q13", 4],
  ["Q14", 4],
  ["Q15", 4],
  ["Q16", 4],
  ["Q17", 4],
  ["Q21", 7],
  ["Q22", 7],
  ["Q23", 7],
  ["Q31", 7],
  ["Q32", 7],
  ["Q34", 7],

  ["Q41", 6],
  ["Q42", 6],
  ["Q43", 6],

  ["Q51", 5],
  ["Q52", 5],
  ["Q53", 5],

  ["Q61", 4],
  ["Q62", 4],

  ["Q71", 4],
  ["Q72", 4],
  ["Q73", 4],

  ["Q81", 7],
  ["Q82", 7],

  ["Q91", 10],
```

I-3- Frontend et résultats:

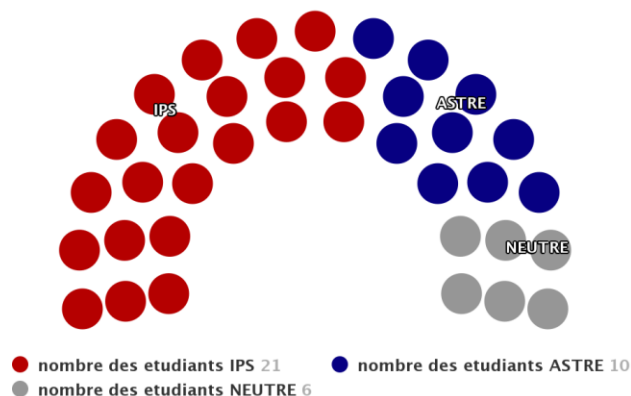
Ceci est la dernière partie avant l'affichage des pages web du projet. Le front est composé du html, css et javascript. Il respecte l'architecture MVC où on trouve les services qui réalisent le calcul des scores de chaque étudiant, les contrôleurs qui gèrent les événements de la vue (html) et faire le binding des données. Le frontend reçoit 2 fichiers JSON générés par le script C#. Le premier contient toutes les hypothèses avec leurs poids et le deuxième contient les réponses de chaque étudiant ainsi que le nombre de points qu'il reçoit à chaque question. Le service calcul score crée une Map qui contient le numéro de l'étudiant comme clé et son score comme valeur. Ce score est calculé en effectuant la correspondance entre les réponses de l'étudiant et les hypothèses et en faisant le produit entre le poids et le nombre des points.

L'utilisateur par la suite utilise les sélectionneurs pour changer les poids des hypothèses et visualiser les résultats sur les deux chartes dans les pages proportions et détails.

```
Map(37) i
▼ [[Entries]]
▶ 0: {"20190448" => -2}
▶ 1: {"190568" => 159}
▶ 2: {"21005534" => 97}
▶ 3: {"191228" => 78}
▶ 4: {"20193694" => 118}
▶ 5: {"2103488" => -76}
▶ 6: {"20191136" => -48}
▶ 7: {"21004647" => -63}
▶ 8: {"21004530" => 53}
▶ 9: {"20192581" => -109}
▶ 10: {"21005983" => 95}
▶ 11: {"190472" => 134}
▶ 12: {"2105428" => -56}
▶ 13: {"21005406" => -5}
▶ 14: {"21003710" => -31}
▶ 15: {"2105955" => -77}
▶ 16: {"20191038" => 38}
▶ 17: {"20195066" => 38}
▶ 18: {"21005457" => -31}
▶ 19: {"21004605" => -48}
▶ 20: {"2105403" => 33}
▶ 21: {"20208092" => -29}
▶ 22: {"2103704" => -48}
▶ 23: {"20191209" => -9}
▶ 24: {"2105425" => 24}
▶ 25: {"2103477" => -104}
▶ 26: {"2104601" => -13}
▶ 27: {"2104614" => 24}
▶ 28: {"2104673" => -55}
▶ 29: {"20190488" => 74}
▶ 30: {"20190829" => 23}
▶ 31: {"2103468" => -104}
▶ 32: {"192010" => 109}
▶ 33: {"20190137" => -114}
```

Les proportions des étudiants

Repartition des étudiants en se basant sur leurs scores



Highcharts.com



II- Projet CMC

Introduction

Le projet CMC consiste à créer des modèles génériques de données à partir d'une base de données SQL. Cette dernière contient les logs d'un outil informatique. Ces logs décrivent les transitions effectuées par les différents utilisateurs ainsi qu'aux manipulations réalisées sur l'ensemble des fichiers de l'outil. Le modèle générique sert à définir une seule structure de stockage et de transfert de données. Ceci facilite l'exploitation des résultats par la suite et la réalisation du calcul pour savoir les étudiants ayant les indicateurs les plus élevés. Ceci dans le but de déterminer si un étudiant aura le profil d'un leader ou non en s'appuyant sur un ensemble.

Afin d'atteindre cet objectif, plusieurs d'étapes s'imposent pour créer un flux clair et logique du projet.

II-1- Modèle générique et extraction des données:

Cette étape consiste à exploiter les données d'une manière complète et exhaustive afin de concevoir un modèle générique qui s'adapte aux éventuels changements de la structure de données. Ceci a été réalisé dans une première approche en utilisant le framework NodeJS. Cependant, les résultats obtenus n'étaient pas satisfaisants comme le framework a une bibliothèque très limitée par rapport à l'extraction des données SQL et l'adapter au modèle spécifié. Pour cette raison, nous avons fait le choix de changer le langage et le framework pour choisir un framework mieux adapté: .NET C#. Les résultats dans cette seconde approche se sont significativement améliorés ainsi que le temps de calcul.

II-2- Moteur de calcul et indicateur :

Notre indicateur choisi est l'étudiant le plus susceptible d'être le chef de projet.

Pour calculer ça on a travaillé sur 3 hypothèses :

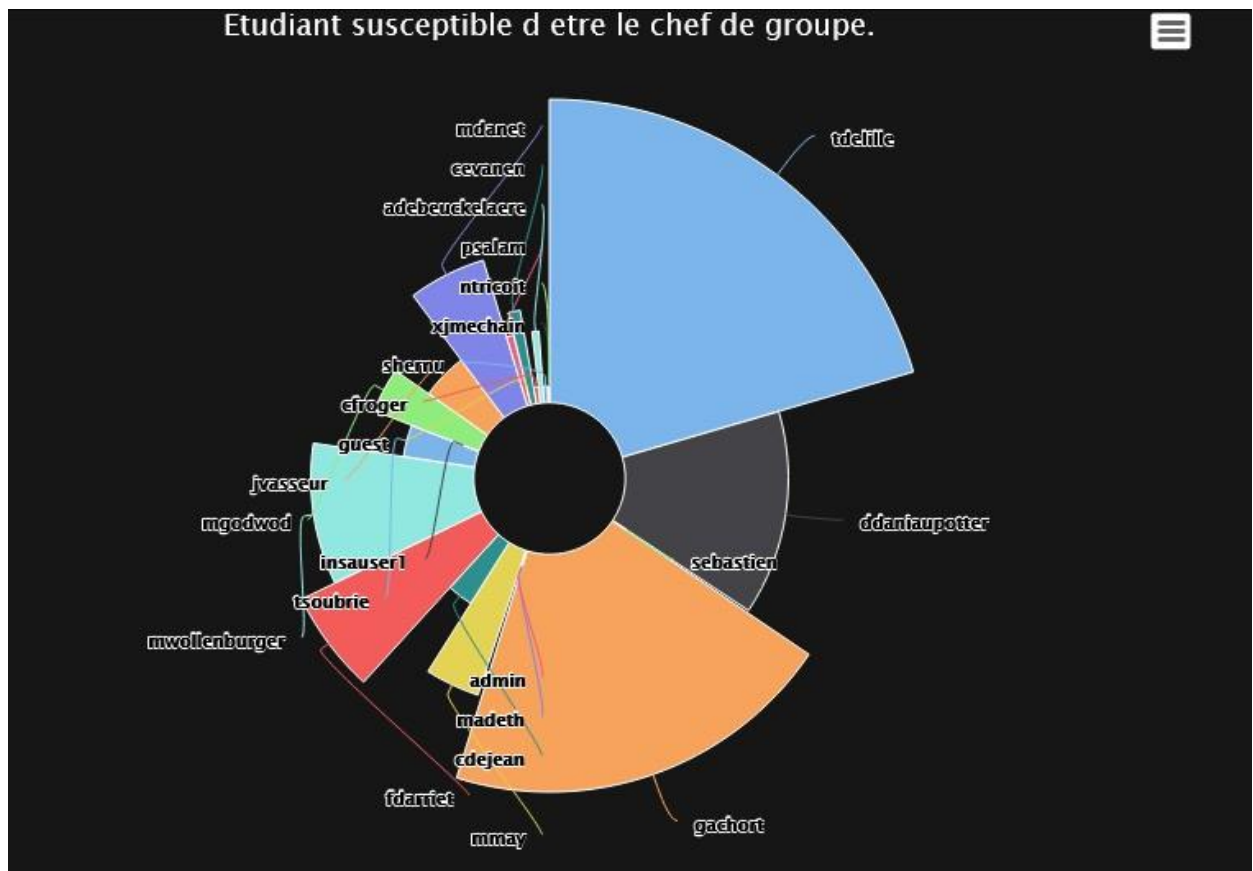
-Sociable : C'est basé sur le nombre de messages portés , le nombre de réponses et ainsi que le ratio Post (calculé par nombre de messages par utilisateur sur le nombre de posts total).

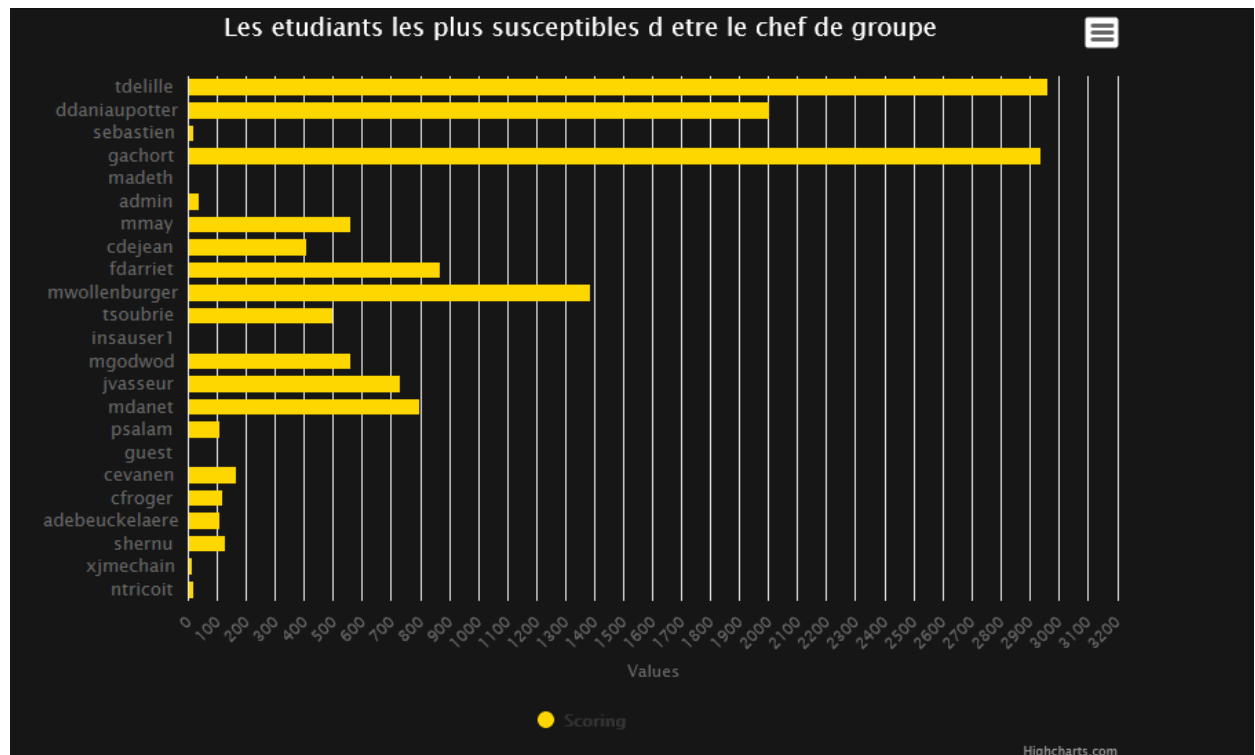
-Actif: C'est basé sur le nombre de cnx , le nombre réponses lus et vus (calculé avec la précision si le délai de message vu est supérieur à 5 s alors c'est considéré comme lu et vu) et le ratio Connexion (pourcentage calculé par le nombre de cnx indiv sur le nombre de cnx total).

-Généreux: C'est basé sur le nombre de fichiers partagés , le nombre de fichiers téléchargés par les autres et le ratio fichiers(pourcentage calculé par le nombre de fichiers téléchargés sur le nombre de fichier téléchargé total).

II-3- Frontend et résultats:

Pour la dernière partie, c'était l'affichage des pages web du projet. Le front est composé du html, css et javascript. Le frontend reçoit un fichier JSON générés par le script .NET C#. Ce fichier json contient le numéro d'étudiant , son score calculé ainsi que les ratios fichiers , ratios postes et les ratios connexion. Avec la bibliothèque Highcharts on a réussi à visualiser les résultats web suivant l'indicateur calculé et les portions.







Structure de données choisie :

```
Users > macbook > Downloads > {} etudiants.json > {} 0 > [ ] transitions
1  [
2    {
3      "id": 0,
4      "pseudo": "tdeille",
5 >   "transitions": [ ...
906  ],
907 >   "fichiers": [ ...
1608  ]
1609  },
1610  {
1611    "id": 1,
1612    "pseudo": "ddaniaupotter",
1613 >   "transitions": [ ...
2514  ],
2515    "fichiers": [
2516      {
2517        "id": 2,
2518        "idforum": 353,
2519        "idMsg": 1530,
2520        "date": "2009-02-27T23:00:00.000Z",
2521        "nbrDownload": 3
2522      },
2523      {
2524        "id": 12,
2525        "idforum": 353,
2526        "idMsg": 1616,
2527        "date": "2009-03-03T23:00:00.000Z",
2528        "nbrDownload": 2
2529      },
2530      {
2531        "id": 20,
2532        "idforum": 332,
2533        "idMsg": 2161,
2534        "date": "2009-03-22T23:00:00.000Z",
2535        "nbrDownload": 0
2536      },
2537      {
2538        "id": 2,
2539        "idforum": 353,
2540        "idMsg": 1530,
2541        "date": "2009-02-27T23:00:00.000Z",
2542        "nbrDownload": 3
2543      },
2544      {
2545        "id": 12,
2546        "idforum": 353,
2547        "idMsg": 1616,
2548        "date": "2009-03-03T23:00:00.000Z",
```