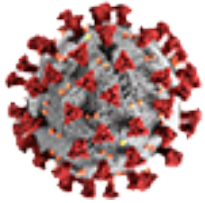


Dossier de Présentation de



coronaReady

Informatique et Science du Numérique
Projet pour le Baccalauréat
(2019-2020)

Dossier fait par : Bechara Rizk (Ts1)

Autres membres du groupe :

- Yara Sabbagha (Ts1)
- Rawi El Khoury (Ts3)
- Sarah El Khazen (Ts2)

Présentation du Projet

Depuis la fin de l'année 2019, le Coronavirus a envahi le monde entier. Sur la majorité de l'internet, l'ampleur qui a été donnée au Coronavirus a généré de l'inquiétude et de la peur. C'est pourquoi, pour notre projet d'ISN, nous avons voulu créer un site internet capable d'aider les gens à passer cette période difficile sans leur faire ressentir beaucoup d'inquiétude tout en donnant des informations correctes et légitimes. Notre projet peut alors être divisé en deux parties.

La première partie consiste uniquement du site web, on celui-ci est composé de plusieurs pages pour diviser les informations d'une façon compréhensible. Tout d'abord on a pour toutes les pages un même entête qui montre notre logo (coronaReady) avec des boutons pour naviguer aux autres pages rapidement. Ensuite il y a une page d'accueil qui contient quelques informations des autres pages pour si quelqu'un voudrait juste s'assurer d'une information commune rapidement. De plus, chaque morceau d'information peut être pressé pour révéler la page qu'il désigne avec le contenu restant. Dans la deuxième page de notre site on présente le virus pour les personnes qui voudrait se cultiver sur l'histoire de ce virus, sa source, les signes d'infection, les symptômes, etc... Dans la troisième page on a mis une multitude d'articles fiables, chacun discutant un aspect du virus et du monde durant cette période. Pour chaque article on peut voir une image provenant du site d'origine, le titre de l'article et un bout de texte décrivant ce dont l'article parle. À tout moment l'utilisateur peut presser sur n'importe quel article pour l'ouvrir dans un nouvel onglet de son moteur de recherche pour lire l'article en entier. Dans la quatrième page on a mis quelques statistiques pour le monde entier et quelques pays spécifiques. On peut voir le nombre de cas total, cas actifs, nombre de mort et nombre de survivants. De plus en bas à droite on a mis la dernière date où ces statistiques ont été renouvelées. Dans les deux pages suivantes on a mis des genres de listes, l'une montrant une liste normale avec des points pour chaque élément de la liste décrivant des conseils pour cette période. La deuxième est une liste avec des carreaux cochables pour chaque élément qui représentent des activités à faire durant cette période. La dernière page représente un quiz que l'utilisateur peut utiliser pour tester ses connaissances sur le virus, or chaque question présente trois réponses possibles avec une d'entre elles correcte que l'utilisateur doit presser. L'usager possède alors dix secondes pour répondre à chaque question et au cas où sa réponse est correcte il aura un cercle vert en bas à droite qui sera rouge au cas où la réponse est erronée. Finalement l'utilisateur aura un score avec une image spécifique dépendamment de ce score.

La deuxième partie de notre projet consiste en des codes pythons qui vont accéder les codes de notre site web pour renouveler les statistiques avec des nombres que nous allons entrer dans une interface

Update coronaReady

Enter The Numbers To Update The Statistics Page

World Total Cases:

World Death Cases:

World Recovered Cases:

World Active Cases:

Lebanon Total Cases:

Lebanon Deaths Cases:

Lebanon Recovered Cases:

Lebanon Active Cases:

Iran Total Cases:

Iran Deaths Cases:

Iran Recovered Cases:

Iran Active Cases:

Japan Total Cases:

Japan Deaths Cases:

Japan Recovered Cases:

Japan Active Cases:

Update Numbers

graphique. L'ordinateur va alors prendre les nombres que nous avons entre pour ensuite modifier le code des pages du site web et aussi renouveler les graphiques en utilisant une base de données externe qui elle-même est renouvelée chaque jour.

Aussi, l'ordinateur va ouvrir automatiquement une page d'un autre site internet pour nous aider à remplir le nombre de cas dans notre interface graphique.

Ceci est une image de notre interface graphique créer en langage python pour renouveler les nombres des statistiques. L'utilisateur va entrer les valeurs dans les cases blanches puis ensuite presser le bouton en bas à droite pour changer les valeurs du code du site internet. En pressant ce bouton l'utilisateur va aussi

initialiser le renouvellement des graphiques grâce à l'aide d'une base de données externe.

coronaReady

Home

Virus Information

People may experience

- Cough
- Fever
- Tiredness
- Difficulty breathing (severe cases)
- Very occasionally, loss of appetite

Statistics

World-wide cases	Lebanese cases
Total: 3,249,764	Total: 725
Active: 2,003,151	Active: 551
Recovered: 1,017,171	Recovered: 150
Deaths: 229,442	Deaths: 24

Articles

Speakup Covid
"With reference to the ethical code of psychologists, this professional and experienced team will be offering psychological support to the frontline Health Caregivers dealing with the outbreak of Novel Coronavirus (COVID-19)."

How The Pandemic Will End ~ The Atlantic
"Three months ago, no one knew that SARS-CoV-2 existed. Now the virus has spread to almost every country, infecting at least 446,000 people whom we know about, and many more whom we do not."

Tips

- Stay Home
- Wash Your Hands
- Practice Social Distancing
- Rest
- Keep Warm

Ceci est la page d'accueil de notre site internet, on peut voir en haut le logo et les boutons avec les boutons de la page même qui un fond plus

fonce que les autres, et en bas on voit les bouts d'information des autres pages du site qui peuvent être presser pour ouvrir la totalité de la page concernée.

Structure Générale de Notre Projet

Comme notre projet est divisé en deux parties -la partie du site web et la partie python-, il n'y a pas qu'un seul algorithme principal mais plutôt deux. Tout d'abord on a la page d'accueil du site web qui peut directement mener aux autres pages grâce à des boutons. Et pour la partie python on a trois fichiers python : un qui s'occupe de renouveler les graphiques, un autre dans lequel nous allons entrer les valeurs pour les nouvelles statistiques, et finalement un troisième qui appelle les méthodes des deux précédents pour les mettre en marche.

```
#!/usr/bin/env python3
```

 Interpréter le fichier en langage python

```
from python_program.updating.update_graphs import Graph
```

 Importer le code qui s'occupe des

```
from python_program.updating.update_numbers import NewNumbers
```

 Importer le code qui s'occupe des valeurs statistiques

```
number=NewNumbers()  
number.title("Update coronaReady")  
number.configure(bg="#cfcfcf")  
number.mainloop()  
lebanon=Graph("Lebanon")  
iran=Graph("Iran")  
japan=Graph("Japan")
```

On appelle les méthodes du deuxième code importé pour ouvrir une fenêtre

On appelle les méthodes importées du premier code qui vont renouveler les graphiques

Code python qui importe et utilise les deux autres codes

```
<head>  
<title>coronaReady - Home</title>  
<meta name="keywords" content="coronaReady, coronavirus, corona, COVID19, COVID-19">  
<meta name="description" content="A page about the coronavirus">  
<meta name="author" content="coronaReady">  
<meta charset="utf-8">  
<link rel="stylesheet" href="website/universal_css.css">  
<link rel="stylesheet" href="website/home_page/home_page.css">  
<link rel="icon" href="https://raw.githubusercontent.com/bechara-rizk/coronaReady/master/python_program/images/corona_logo.png">  
</head>  
<body>  
<div class="header">  
  
<a href="index.html"><button class="head_buttons head_buttons_active", title="Go to Home page"></button></a>  
<a href="info_page.html"><button class="head_buttons", title="Go to Informations page"></button></a>  
<a href="articles_page.html"><button class="head_buttons", title="Go to Articles page"></button></a>  
<a href="stats_page.html"><button class="head_buttons", title="Go to Statistics page"></button></a>  
<a href="tips_page.html"><button class="head_buttons", title="Go to Tips page"></button></a>  
<a href="sum_page.html"><button class="head_buttons", title="Go to Daily Summary page"></button></a>  
<a href="quiz_page.html"><button class="head_buttons", title="Go to Quiz page"></button></a>  
</div>
```

Code HTML qui importe les fichiers CSS (lignes 7 et 8) et contient les boutons pour naviguer aux autres pages (lignes 14 jusqu'à 20)

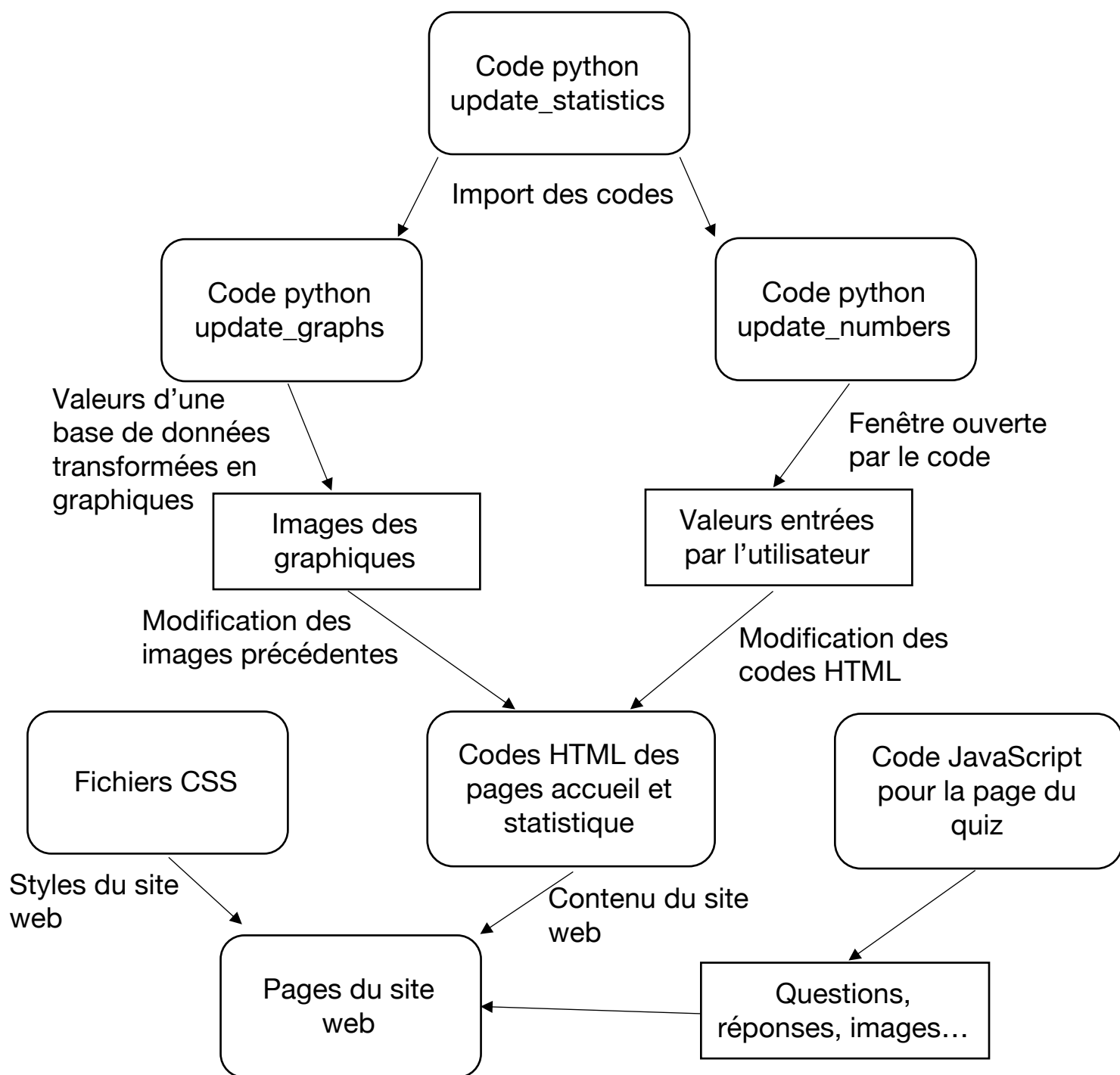


Schéma expliquant les interactions au sein de nos codes

Répartition des Tâches au Sein de Notre Groupe

Comme notre projet était composé de quatre langages de code, il était nécessaire de bien se diviser les tâches pour qu'on puisse achever celui-ci. Tout d'abord, chaque site web commence par un design qui est censé montrer déjà une image de comment le site web va être. Nous avons alors choisi Sarah et moi pour faire cette partie du projet. Après cela nous devions encore se diviser les tâches pour la « construction » du site web, en effet il est composé de plusieurs pages. Par conséquent, on a donné à chaque membre du groupe des pages du site web à coder. Sarah était en charge de faire les deux premières pages du site web, c'est à dire la page d'accueil et la page d'information du virus. Rawi devait s'occuper des pages d'articles et de statistiques du site web. Yara a codé les pages des conseils et des activités journalières. Donc, j'ai pris en charge de m'occuper de la page du quiz. Chaque page nécessitait au moins deux fichiers, un fichier HTML pour le contenu de la page et un fichier CSS pour le style et la façon dont ce contenu est vu. La page du quiz était une exception à ceci puisqu'en plus des deux autres fichiers elle avait besoin d'un fichier JavaScript pour pouvoir faire marcher le quiz. Finalement, Yara et Rawi ont cherché les textes et le contenu du web site qui allaient être mis dans les pages.

De plus, Yara et moi avons travaillé sur la partie python de notre projet qui s'occupait de renouveler les statistiques. Yara s'est plus intéressée sur la partie où l'algorithme importait les valeurs de la base de données et comment les arranger en une liste puis je me suis occupé de la partie interface graphique de notre projet où l'utilisateur peut entrer les nouvelles valeurs pour que la partie écrite par Yara puisse les mettre dans le code HTML et j'ai aussi pris en charge la transformation des listes de valeurs trouvées par l'algorithme de Yara en graphiques pour ensuite les sauvegarder en images.

Exemples de Parties de Code et Difficultés

La partie de notre projet que j'ai trouvée la plus difficile était de faire le code JavaScript qui gère la page du quiz. En effet il n'est pas possible de coder les fonctionnalités de notre quiz avec HTML et CSS uniquement donc on a utilisé JavaScript pour pouvoir changer les questions, s'assurer de la réponse choisie, utiliser un minuteur, et attribuer un score à l'utilisateur. De plus j'ai personnellement choisi de faire cette partie dans le but de commencer à apprendre ce langage (puisque avant notre projet je n'y avais jamais touché). Par ailleurs je vais donc expliquer quelques exemples de code tirés de notre fichier JavaScript.

Tout d'abord il est essentiel de marquer que contrairement à Python, dans JavaScript nous devons finir chaque ligne par un point-virgule (;).

```
2  const start = document.getElementById("start");
3  const quiz = document.getElementById("quiz");
4  const question = document.getElementById("question");
5  const qImg = document.getElementById("qImg");
6  const choiceA = document.getElementById("A");
7  const choiceB = document.getElementById("B");
8  const choiceC = document.getElementById("C");
9  const counter = document.getElementById("counter");
10 const timeGauge = document.getElementById("timeGauge");
11 const progress = document.getElementById("progress");
12 const scoreDiv = document.getElementById("scoreContainer");
```

Ici premièrement nous définissons des variables qui « accèdent » au code HTML grâce au ID spécifier dans celui-ci pour changer leurs attributs. Nous pouvons définir des variables grâce à « const » (le type de la

variable devra rester le même) ou aussi avec « let » (ou le type peut être modifié).

```
15 let questions = [
16   {
17     question : "How can you contract the COVID-19? ",
18     imgSrc : "python_program/images/corona_logo.png",
19     choiceA : "Through animals",
20     choiceB : "Through droplets of saliva or discharge from the nose",
21     choiceC : "Through heated food",
22     correct : "B"
23   },
```

Ici on a défini une liste imbriquée dans une autre liste. Chaque élément de la liste primaires contient des éléments qui définissent pour chaque question du

quiz : la question, les réponses possibles avec la bonne réponse et une image pour chaque question.

```
77 function renderQuestion(){
78   let q = questions[runningQuestion];
79   question.innerHTML = "<p>"+ q.question + "</p>";
80   qImg.innerHTML = "";
81   choiceA.innerHTML = q.choiceA;
82   choiceB.innerHTML = q.choiceB;
83   choiceC.innerHTML = q.choiceC;
84 }
```

Ici on définit une fonction qu'on va appeler prochainement qui accède au code HTML pour ajouter ou changer la question à chaque fois ou la fonction est appelée.

```
86 start.addEventListener("click",startQuiz);
87 function startQuiz(){
88   start.style.display = "none";
89   renderQuestion();
90   quiz.style.display = "block";
91   renderProgress();
92   renderCounter();
93   TIMER = setInterval(renderCounter,1000);
94 }
```

Dans cette partie la fonction définie sera appelée quand l'élément du code HTML portant l'ID « start ». La fonction va alors cacher le bouton qui initialise le quiz

(ligne 18), montrer la partie qui contient les éléments de la question, et appeler des fonctions pour faire fonctionner le quiz.

```
153 function scoreRender(){
154     scoreDiv.style.display = "block";
155     const scorePerCent = Math.round(100 * score/questions.length);
156     let img = (scorePerCent >= 80) ? "python_program/images/result_img_5.png" :
157             (scorePerCent >= 60) ? "python_program/images/result_img_4.png" :
158             (scorePerCent >= 40) ? "python_program/images/result_img_3.png" :
159             (scorePerCent >= 20) ? "python_program/images/result_img_2.png" :
160             "python_program/images/result_img_1.png";
161     scoreDiv.innerHTML = "";
162     scoreDiv.innerHTML += "<p>"+ scorePerCent +"%</p>";
163 }
```

Et finalement dans cette partie on a défini une fonction qui sera appelée à la fin du quiz pour montrer son score à l'utilisateur. Tout d'abord elle montre la partie qui contient les éléments du

score, calcul celui-ci en un pourcentage par rapport au score obtenu sur le nombre de questions, choisi une image dépendamment de ce calcul et puis ajoute l'image et le pourcentage au code HTML.

Problèmes qui Demeurent et Améliorations Possibles

Premièrement un des problèmes que nous avons rencontré était que nous avons changé la composition de notre projet une multitude de fois. En effet, nous voulions tout d'abord faire une application iOS (pour téléphone) avec une application python (pour ordinateur). Cette idée a été changée puisque l'application iOS allait consommer trop de temps pour la faire et nous avons alors choisi de faire un site web python mais par défaut d'explications trouvées nous ne savions pas vraiment beaucoup comment programmer ce site web. Finalement nous avons choisi de faire un site web utilisant HTML, CSS et JavaScript en plus d'une application python qui va renouveler les valeurs de notre site web plus facilement que si on le faisait totalement manuellement.

Un des problèmes que nous avons rencontrés était que les tailles utilisées dans notre site web étaient uniquement faites pour une seule résolution d'écran et que si on l'ouvrait sur un autre appareil le design de notre site web allait devenir faux et avait besoin d'arrangement. Une solution possible à cela était de refaire tout le style du site web prenant en compte une méthode appelée « Responsive Web Design ». Cette méthode nous aurait permis de construire le site de plusieurs façons chacune appartenant à un intervalle de résolution d'écrans pour que le contenu de chaque page reste lisible.

De plus, dans la partie créant les graphiques qui montrent les cas de chaque pays on a eu quelques problèmes car l'axe des cas montrait toutes les valeurs ce qui créait un effet de désordre dans les graphiques.

Un autre problème qu'on a rencontré était que pour renouveler les statistiques on devait faire cela manuellement avec l'aide d'un autre site web. On aurait voulu que les nombres se renouvèlent seuls automatiquement. Une amélioration aurait été que notre code accède le code de l'autre site web pour renouveler les statistiques seul.

Impressions Personnelles sur le Travail et Conclusion

Personnellement j'ai aimé travailler sur ce projet puisque j'ai pu appliquer mes connaissances en HTML et CSS, j'ai eu la possibilité de fortifier mon expérience sur la programmation d'une interface graphique simple grâce à Python et le plus important et que j'ai commencé à apprendre un nouveau langage très utile qui est JavaScript.

D'un côté global, je considère le travail effectué par tout le groupe comme complet et bien fait puisque les résultats obtenus sont quelque peu parfaits néanmoins avec quelques fonctionnalités manquantes sur lesquelles j'y travaillerais pour améliorer ce projet et peut être l'utiliser dans une fiche de travaux faits.

Bibliographie et Ressources Extérieures

Lien du Site Internet	Description
https://bechara-rizk.github.io/dossier_projet_isn	Lien du site internet de mon dossier
https://stackoverflow.com	Site internet sur lequel on peut poster notre code au cas où on a un problème qu'on n'arrive pas à résoudre et pour voir des solutions pour les problèmes d'autres utilisateurs.
https://www.geeksforgeeks.org	Site internet expliquant des syntaxes de code.
https://www.w3schools.com	Site internet permettant d'apprendre une multitude de langages de code comme Python, HTML, CSS et JavaScript.
https://github.com	Site internet montrant les codes des projets de beaucoup d'autres personnes pour s'inspirer et s'aider dans le nôtre.
https://www.worldometers.info/coronavirus	Lien du site internet ouvert par le code python pour aider à renouveler le nombre de cas.
https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv	Lien de la base de données utilisée par notre code pour renouveler les graphiques.
https://bechara-rizk.github.io/coronaReady	Lien du « dépôt » (repository) de notre projet sur GitHub.
https://github.com/bechara-rizk/dossier_projet_isn	Lien du « dépôt » (repository) de mon dossier sur GitHub.
https://bechara-rizk.github.io/coronaReady	Lien du site internet de notre projet.