CheckSanté

Les membres

Ophélie Schauffler
Nathan Quellec
Yann Lemen
Léa Habert
Hariprasad Raypoulet

Lien de notre espace Notion :
https://typical-cork-25a.notion.site/CheckSant-b2ab2b102fa041fdaf802b74550a25cc

I. Présentation générale

Fonctionnalités attendues

Principales :

- Proposer au patient de constituer un dossier en ligne avec ses renseignements
- Exposer des modules de prédiction de maladie
- Afficher un graphe sur l'état de santé du patient (sous forme de graphique de type étoile)
- Faire un appel à un calendrier pour enregistrer un rendez-vous.
- Garder, en base de données, l'historique des différents résultats pour le patient et lui permettre d'avoir un suivi de l'évolution de sa santé

Facultatives :

- Faire des suggestions (alimentation, habitudes etc) au patient (ML - classification - aussi)
- Rediriger vers des professionnels de la santé dans la région en fonction de la région (Doctolib et/ou Google)

Différentiel par rapport à l'existant

Il existe des plateformes (Apple health ou Huawei Health) qui utilisent les données des montres connectées ou bien du compte du client afin de faire de la visualisation de données et proposer différentes features.
Nous voulons nous différencier en proposant une plateforme permettant à l'utilisateur de visualiser de façon simple son état de santé global (risque de contracter une maladie, selon ses données de santé), prendre rendez-vous chez un médecin, bénéficier de différents conseils ciblés sur sa santé.
Contrairement à la plateforme Doctolib qui consiste principalement à de la prise de rendez-vous, nous allons suggérer la prise de rendez-vous de façon intelligente en fonction de l'analyse du ou des modèle(s).

Enfin, par rapport à la plateforme d'Améli à venir, "Mon Espace Santé", nous proposons plus que simplement stocker et restituer des informations sur le patient. Nous proposons une première analyse et une interprétation des données.

Personas

Persona 1 : Jeune

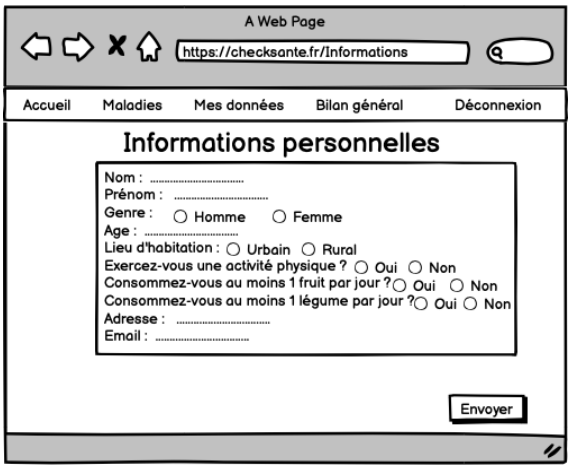
- 18-35 ans
- porté sur la technologie

Persona 2 : Médecin généraliste

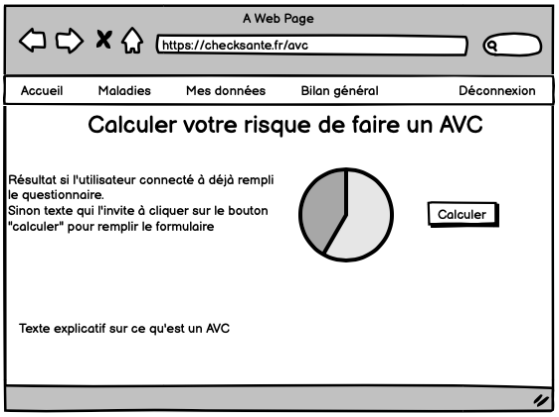
Wireframe

La page d'accueil : Cette page est composée d'une navbar permettant de naviguer dans l'outil, ainsi qu'un texte d'accueil expliquant le but du service.

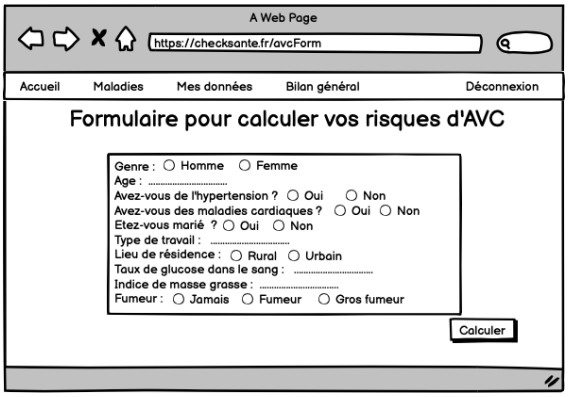
Le formulaire d'informations personnelles : Ce formulaire sert à recueillir les informations générales de l'utilisateur. Les données utiles pour la prédiction d'une maladie seront récupérées dans le formulaire concerné.



La page d'une maladie : Chaque page de maladie sera construite de la même manière. Elle aura un titre indiquant le type de maladie concerné, un texte et un graphique si l'utilisateur a déjà rempli le formulaire de calcul. Un bouton permettra d'accéder au formulaire de saisi des données pour prédire son risque



Le formulaire de calcul de risque d'AVC (ou autre maladie) : Chaque maladie aura un formulaire permettant à l'utilisateur de renseigner les informations nécessaire à la prédiction puis sera renvoyé à la page précédente sur laquelle ses résultats s'afficheront.



Scénario

Scénario 1 : Un nouvel utilisateur rempli sa fiche "informations personnelles"

1. Ouvrir le site
2. Cliquer sur "mes données"
3. Remplir le formulaire
4. Cliquer sur "enregistrer"

Scénario 2 : Tester son risque d'AVC (ou autre)

1. Ouvrir le site
2. Cliquer sur Maladies, AVC
3. Cliquer sur le bouton "Calculer"
4. Remplir le formulaire et cliquer sur "calculer"
5. Visualiser le résultat

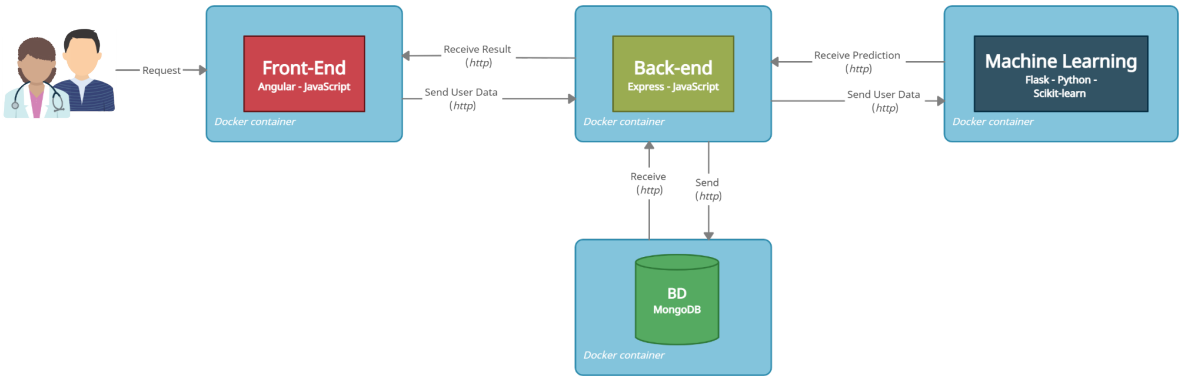
Scénario 3 : Regarder son bilan général

1. Ouvrir le site
2. Cliquer sur “Bilan général”
3. Visualiser le bilan

II. Architecture

1. Agencement global

L’architecture de notre application sera en client-serveur avec un front-end, back-end, composant ML et une base de données conteneurisés avec **Docker**.



Étant donné que notre produit peut évoluer en devant gérer, dans le futur, des données non structurées (imagerie médicale), nous avons fait le choix d’utiliser un moteur de base de données NoSQL : **MongoDB**. De plus, cette base de données accueillera les données de compte d'un utilisateur. En effet, la création d'un compte est cruciale dans la mesure où nous pourrons sauvegarder des informations telles que l'année de naissance, la taille, l'adresse etc sans avoir à redemander à chaque traitement.

Pour les autres frameworks, nous nous référons à la stack **MEAN** (Mongo, Express, Angular, NodeJS), qui à l’avantage d’utiliser JavaScript comme unique langage de programmation. Cela nous permettra de itérer plus rapidement pour réaliser notre Proof Of Concept. Le composant ML, quant à lui, sera codé en Python pour pouvoir utiliser la bibliothèque scikit-learn et pandas, entre autres.

2. Architecture de la brique de Machine Learning

Afin d’exposer les services proposés par la brique de Machine Learning du projet Check_Santé, nous avons fait l’effort de mettre en place une architecture modulable, respectant les standards de PEP 8 (pour les bonnes pratiques de Python).
Pour ce faire, dans le dossier, routes, se trouve les fichiers accueillant les urls d’appel aux services exposés. Pour lancer le module de manière indépendante,

sans démarrer tout le projet, il suffit de se placer dans le dossier machine_learning et d'exécuter la commande :

> python manage.py run

Voici le détail de l'architecture :

```
machine_learning
    ├──brique_ML
        ├── ML_models          # Folder with ML models saved as .joblib
        ├── routes             # Folder with routes towards exposed services
        ├── config             # For configuration
        ├──__init__.py         # Starting point of package brique_ML
    ├── brique_ML.log          # Log file of module
    ├── Dockerfile             # For docker management
    ├── manage.py              # File to launch the module
```

3. Architecture du Back-End

4. Architecture du Front-End

SI externes

Nous prévoyons de faire un appel à un calendrier Google pour enregistrer un rendez-vous chez un médecin en cas de maladie.

Nous pourrions également nous connecter à une API de Google permettant de nous donner les médecins autour d'une localisation donnée (pas possible avec Doctolib car ils ne mettent pas d'API à disposition).

III. Machine Learning

Nous allons utiliser 5 datasets servant à analyser l'état de santé d'un utilisateur.
Voici les datasets dont nous disposons :

1. Masse grasse (body fat)
2. Diabète
3. AVC (stroke)
4. Maladies cardiaques (Heart Failure)
5. Hépatite C

Nous allons, grâce à ces données, calculer le risque de l'utilisateur d'avoir chacune des maladies citées ci-dessus, en fonction des données saisies par l'utilisateur. Nous pourrons ensuite effectuer un traitement selon les résultats obtenus.

Description des datasets

1. Masse grasse

https://www.kaggle.com/fedesoriano/body-fat-prediction-dataset

Liste des estimations du pourcentage de graisse corporelle déterminées par le calcul de la densité corporelle et les mensurations pour 252 hommes.

Nous avons dans ce dataset 252 lignes et 15 colonnes (dont une qui est la variable à prédire “BodyFat”). L’annexe n°1 présente le détail des variables du dataset.

Nous allons donc faire de la régression (apprentissage supervisé) afin de ressortir un score caractérisant le taux de graisse d'une personne.

2. Diabète

https://www.kaggle.com/alexteboul/diabetes-health-indicators-dataset?select=diabetes_012_health_indicators_BRFSS2015.csv

Dataset nettoyé de 253 680 réponses à une enquête du CDC. La variable cible Diabetes_012 comporte 3 classes :

- Absence de diabète ou uniquement pendant la grossesse - 0
- Prédiabète - 1
- Diabète - 2

Nous avons dans ce dataset 253,680 lignes et 22 colonnes (dont une qui est la variable à prédire “Diabetes_012”). L’annexe n°2 présente le détail des variables du dataset.

Nous allons faire de la régression (apprentissage supervisé) afin de faire ressortir un score permettant d'estimer si une personne est diabétique ou pas.

3. AVC

https://www.kaggle.com/fedesoriano/stroke-prediction-dataset

Ce dataset est utilisé pour prédire si un patient est susceptible de subir un accident vasculaire cérébral (AVC) en fonction de paramètres d'entrée tels que le sexe, l'âge, diverses maladies et le tabagisme. Chaque ligne des données fournit des informations pertinentes sur le patient.

Nous avons dans ce dataset 5110 lignes et 12 colonnes (dont une qui est la variable à prédire “Stroke”). L’annexe n°3 présente le détail des variables du dataset.

Nous allons faire de la régression (apprentissage supervisé) afin de donner un score de probabilité caractérisant le risque d'AVC

4. Maladies cardiaques

https://www.kaggle.com/fedesoriano/heart-failure-prediction

Ce dataset a été créé en combinant différents datasets déjà disponibles indépendamment mais non combinés auparavant. Dans ce dataset, 5 datasets sur les maladies cardiaques sont combinés sur 11 caractéristiques communes.

Nous avons dans ce dataset 918 lignes et 12 colonnes (dont une qui est la variable à prédire “HeartDisease”). L’annexe n°4 présente le détail des variables du dataset.

Nous allons faire de la régression (apprentissage supervisé).

5. Hépatite C

https://www.google.com/url?q=https://www.kaggle.com/fedesoriano/hepatitis-c-dataset&sa=D&source=docs&ust=1646936507698945&usg=AOvVaw23vrixdy8GBZeuTrFvpHsb

Ce dataset contient les valeurs de laboratoire des donneurs de sang et des patients atteints d'hépatite C ainsi que des valeurs démographiques comme l'âge.
Les individus sont catégorisées de la façon suivante :

- Blood-donor - 0
- Suspect blood-donor (risque hépatique) - 0s
- Hepatitis - 1
- Fibrosis - 2
- Cirrhosis - 3

Nous avons dans ce dataset 615 lignes et 14 colonnes (dont une qui est la variable à prédire “Category”). L’annexe n°5 présente le détail des variables du dataset.

Nous allons faire de la régression (apprentissage supervisé).

Pour les données sur AVC et diabète, nous pourrons également faire du clustering. De cette façon, nous pourrons donner des recommandations de changement dans le mode de vie de l’utilisateur pour qu’il réduise son risque. Certaines variables sont présentes dans plusieurs datasets, cf annexe n°6.

Personas de test

Persona 1 : Jeune actif

- Bonne hygiène de vie
- Attendu : pas de risque de maladie

Persona 2 : Adulte surpoids

- Pas d’activité physique

- Mauvaise alimentation
- Attendu : risque élevé de maladie

IV. Annexes

Annexe n°1 : Les variables du dataset Masse grasse

1. Density determined from underwater weighing
2. Percent body fat from Siri's (1956) equation
3. Age (years)
4. Weight (lbs)
5. Height (inches)
6. Neck circumference (cm)
7. Chest circumference (cm)
8. Abdomen 2 circumference (cm)
9. Hip circumference (cm)
10. Thigh circumference (cm)
11. Knee circumference (cm)
12. Ankle circumference (cm)
13. Biceps (extended) circumference (cm)
14. Forearm circumference (cm)
15. Wrist circumference (cm)

Annexe n°2 : Les variables du dataset Diabète

1. diabète : 0 = no diabetes 1 = prediabetes 2 = diabetes
2. highBP : 0 = no high BP 1 = high BP (binaire)
3. HighCol : 0 = no high cholesterol 1 = high cholesterol (binaire)
4. CholCheck : 0 = no cholesterol check in 5 years 1 = yes cholesterol check in 5 years (binaire)
5. BMI
6. Smoker : Have you smoked at least 100 cigarettes in your entire life? [Note: 5 packs = 100 cigarettes] 0 = no 1 = yes (binaire)
7. Stroke : (Ever told) you had a stroke. 0 = no 1 = yes (binaire)
8. HeartDiseaseOrAttack : coronary heart disease (CHD) or myocardial infarction (MI) 0 = no 1 = yes (binaire)
9. PhysActivity : physical activity in past 30 days - not including job 0 = no 1 = yes (binaire)
10. Fruits : Consume Fruit 1 or more times per day 0 = no 1 = yes (binaire)
11. Veggies : Consume Vegetables 1 or more times per day 0 = no 1 = yes (binaire)

12. HyvAlcohol : Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week) 0 = no 1 = yes (binaire)
13. HealthCare : Have any kind of health care coverage, including health insurance, prepaid plans such as HMO, etc. 0 = no 1 = yes (binaire)
14. NoDocBcCost : Was there a time in the past 12 months when you needed to see a doctor but could not because of cost? 0 = no 1 = yes (binaire)
15. GenHealth : Would you say that in general your health is: scale 1-5 1 = excellent 2 = very good 3 = good 4 = fair 5 = poor
16. MentalHealth : Now thinking about your mental health, which includes stress, depression, and problems with emotions, for how many days during the past 30 days was your mental health not good? scale 1-30 days
17. PhysHealth : Now thinking about your physical health, which includes physical illness and injury, for how many days during the past 30 days was your physical health not good? scale 1-30 days
18. Diff walk : Do you have serious difficulty walking or climbing stairs? 0 = no 1 = yes (binaire)
19. Sex : 0 = female 1 = male (binaire)
20. Age : 13-level age category (_AGEG5YR see codebook) 1 = 18-24 9 = 60-64 13 = 80 or older
21. Education : Education level (EDUCA see codebook) scale 1-6 1 = Never attended school or only kindergarten 2 = Grades 1 through 8 (Elementary) 3 = Grades 9 through 11 (Some high school) 4 = Grade 12 or GED (High school graduate) 5 = College 1 year to 3 years (Some college or technical school) 6 = College 4 years or more (College graduate)
22. Income : Income scale (INCOME2 see codebook) scale 1-8 1 = less than $10,000 5 = less than $35,000 8 = $75,000 or more

Annexe n°3 : Les variables du dataset AVC

1. id: unique identifier
2. gender: "Male", "Female" or "Other"
3. age (years)
4. hypertension: 0 = no hypertension, 1 = hypertension (binaire)
5. heart_disease: 0 = no heart diseases, 1 = heart disease (binaire)
6. ever_married: "No" or "Yes"
7. work_type: "children", "Govt_jov", "Never_worked", "Private" or "Self-employed"
8. Residence_type: "Rural" or "Urban"
9. avg_glucose_level: average glucose level in blood
10. bmi: body mass index
11. smoking_status: "formerly smoked", "never smoked", "smokes" or "Unknown"*
12. stroke: 1 = the patient had a stroke, 0 = not

Annexe n°4 : Les variables du dataset Maladies cardiaques

1. Age: age of the patient [years]
2. Sex: sex of the patient [M: Male, F: Female]
3. ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
4. RestingBP: resting blood pressure [mm Hg]
5. Cholesterol: serum cholesterol [mm/dl]
6. FastingBS: fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
7. RestingECG: resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]
8. MaxHR: maximum heart rate achieved [Numeric value between 60 and 202]
9. ExerciseAngina: exercise-induced angina [Y: Yes, N: No]
10. Oldpeak: oldpeak = ST [Numeric value measured in depression]
11. ST_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]
12. HeartDisease: output class [1: heart disease, 0: Normal]

Annexe n°5 : Les variables du dataset Hépatite C

Attributes 1 to 4 refer to the data of the patient:

1. X (Patient ID/No.)
2. Category (diagnosis) (values: '0=Blood Donor', '0s=suspect Blood Donor', '1=Hepatitis', '2=Fibrosis', '3=Cirrhosis')
3. Age (in years)
4. Sex (f,m)

Attributes 5 to 14 refer to laboratory data:

5. ALB
6. ALP
7. ALT
8. AST
9. BIL
10. CHE
11. CHOL
12. CREA
13. GGT
14. PROT

Annexe n°6 : Nombre d'occurrence par variables redondantes

Âge	5
Gender	4
Cholestérol	3
BMI	2
Stroke	2
Heart Disease	3