Une image contenant conception, capture d’écran, Graphique, Bleu électrique

Description générée automatiquementUne image contenant Graphique, graphisme, Caractère coloré, Police

Description générée automatiquement

Conception des interfaces web

Lot 1 – Mai 2023

**Patch Services**

**Développements**

# Objet

## Interfaces

Les interfaces sont celles que verront les utilisateurs que ce soient les clients et les équipes de support AC Engineering

Elles sont accessibles avec un navigateur web type Chrome, Firefox, Edge et Safari (Mac OS)

Leur usage ne sera pas bloqué pour des tablettes et smartphones mais sur de petits écrans et sans souris l’usage peut être plus difficile et moins convivial. Une version ultérieure adaptée pourra être envisagée mais ne fait pas partie du lot 1.



# Principe de construction

Donné pour information le principe de construction des pages permet de mieux comprendre ce qui est possible et rapide à faire et quelles sont les limites.

Les pages sont construites page par page avec une navigation possible entre elles et des contraintes de sécurité d’accès.

## Le contenu général de la page

La page est écrite en **Html** : cela permet de donner les éléments présents

A ce stade le look est très basique, il n’y a :

* Pas de charte graphique
* Pas de gadgets ou de composants graphiques évolués
* Pas de gestion de la langue
* Pas de données dans la vue (comme par exemple la liste des fichiers traités ou à télécharger).

Exemple d’illustration

Une image contenant texte, capture d’écran, Police, ligne

Description générée automatiquement

## Ajout de feuilles de style

Utilisation de CSS pour ajouter des feuilles de style et donner la **charte graphique** :

* Choix des positions, des alignements, des polices, des couleurs
* Ajout d’images et de trames de fond

Exemple d’illustration :

Une image contenant texte, ordinateur, capture d’écran, Site web

Description générée automatiquement Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

* Pour les **icones** il y a l’excellent font-awesome qui propose un très grand nombre d’icônes « cohérentes » c’est-à-dire qu’elles sont assez homogènes et sobres pour un usage pro.

<https://fontawesome.com/icons>

Un gros plus est le moteur de recherche : Avec des variantes d’épaisseur. (La couleur se règle avec le CSS précédent)

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

Il suffit de bien noter le code écrit sous l’image pour pouvoir l’utiliser facilement dans l’application dans une barre d’outils, un bouton, un texte….

Autres exemples : <https://primeng.org/icons>

## Un thème = La charte graphique

Pour une expérience utilisateur conviviale il ne suffit pas de pouvoir changer la couleur d’un bouton, il faut une cohérence d’ensemble : on n’écrit pas en bleu foncé sur un fond noir.

Les possibilités en termes de charte graphique sont infinies mais il est plus simple de s’inspirer de thèmes classiques (avec ensuite quelques variations au besoin)

Un thème est un ensemble de CSS cohérent qui a un rendu agréable sur différents contenus (boutons, titres, etc …)

Quelques exemples ici pour illustrer avec une large partie des boutons présentés dans une seule page unique pour voir le rendu possible. NB :il y en a bcp d’autres.

Exemples de thèmes qui montrent tous les boutons et listes : un même contenu mais avec des thèmes variables.

<https://designer.primeng.org/#/>

Les thèmes ne sont pas figés on peut bien sur les faire varier ensuite au cas par cas : c’est un modèle de fond un peu comme un template word ou PowerPoint.

NB : les thèmes Dark sont foncés

Une image contenant capture d’écran, texte, logiciel, Logiciel multimédia

Description générée automatiquement

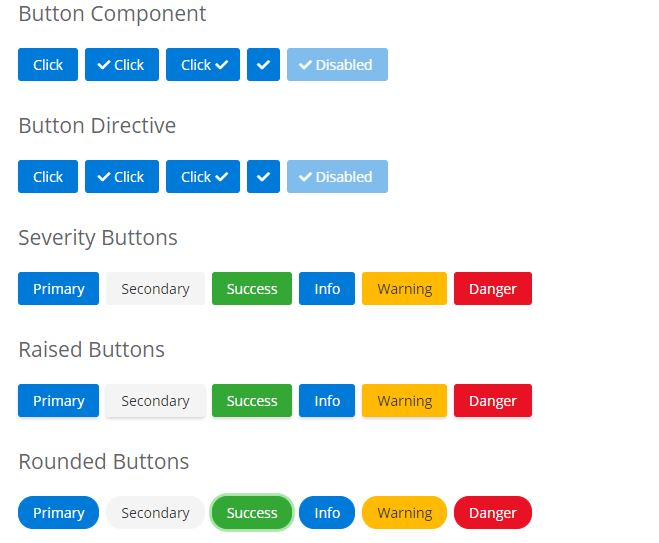
Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

NB : un thème peut être modifié ultérieurement sans ‘casser’ le fonctionnement de l’applicatif.

Exemple Les boutons peuvent être carrés (cf ci-dessus) ou arrondis avec effet plat ou ombré avec ou sans icone.

Ex :



## La gestion des langues

Pour gérer le français et l’anglais, l’intégralité du texte affiché à l’écran est retirée de la page html au profit d’une codification. Ces codes, ici en majuscules ne sont en fait jamais affichés.

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Une Directive | **translate** va substituer la clé par la langue de l’utilisateur connecté si disponible, ou à défaut l’anglais.

La traduction des clés est donnée dans un fichier à part et il n’est pas besoin d’être développeur pour écrire ou modifier les clés ou de rajouter des langues.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Pour faire une version en anglais :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

# Dynamique des pages

A ce stade on peut avoir des pages « jolies » mais elles ne font rien : comme des pages d’un document ou les pages de thèmes ci-dessus. Vous pouvez cliquer sur les boutons mais rien ne se produit et rien n’est affiché.

Il faut ajouter la logique du code de l’application et l’envoi + la récupération de données.

Le langage utilisé est **Angular** (c’est du TypeScript -> Javascript) qui est compris pas les navigateurs web qui s’appuie sur du code métier **C#** sur le serveur.

C’est une des étapes les plus complexes mais nous pouvons nous appuyer sur de composants existants pour ne pas réinventer la roue à chaque bouton ;

La page de thème n’est pas exhaustive mais donne un aperçu des gadgets disponibles à composer

<https://designer.primeng.org/#/>

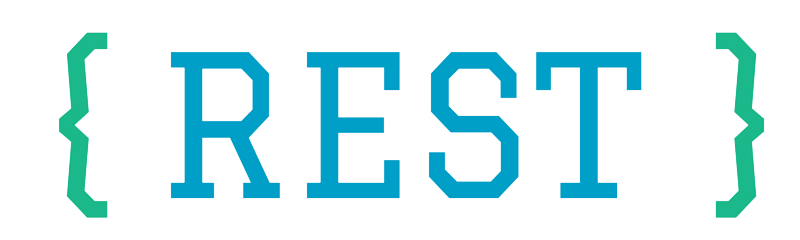
Il y en a bien d’autres ici : mais il faut les cliquer un par un dans Components à gauche

<https://primeng.org/selectbutton>

# Données

Les pages web ne contiennent aucune donnée sauf celles saisies par les utilisateurs sur leur écran.

Pour avoir du contenu comme « la liste des de demandes client à traiter » ou état de de mon ticket en cours de traitement » ou « téléchargement du fichier », e code Angular s’appuie sur un développement en **C#** qui contient toute la logique métier.

Technologie utilisée : 

Une image contenant texte, capture d’écran, diagramme, Rectangle

Description générée automatiquement

Les données persistantes comme l’historique des traitements, les comptes utilisateurs sont stockés dans une base de données type MySQL ou PostgresSQL

# Sécurité

## Sécurité de la connexion

La connexion à l’application se fait via un classique login / mot de passe

Le login sera l’email de l’utilisateur

Le mot de passe devra être forcé à un niveau minimal de complexité (ex : 8 chars, minuscule, majuscule, et un char spécial).

Sans que ce soit obligatoire je suggère de demander en plus le compte client au login : celui que vous avez créé quand il a souscrit l’abonnement

En effet : l’email n’est pas forcément en relation avec le compte client (ex @orange.fr @gmail.com) et rien n’interdit potentiellement à un même utilisateur avec le même email de traiter sur deux comptes abonnés.

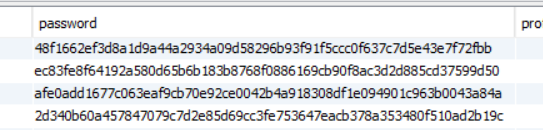
Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

Le mot de passe est masqué.

L’envoi est sécurisé via Https (voir ci-dessous)

Les mots de passes ne sont pas stockés en clair en base de données : on ne garde que leur empreinte SHA-256 ou > + Salt



En pratique il est impossible de récupérer le mot de passe original.

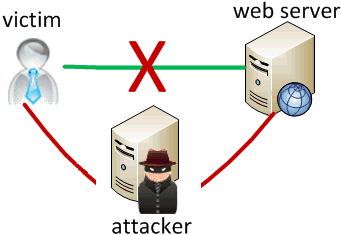
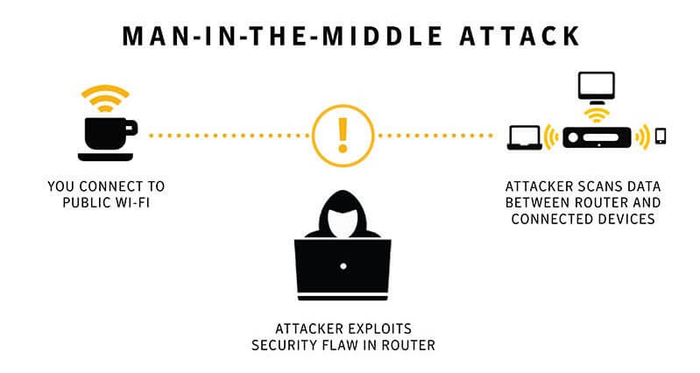
Quand on vérifie si l’utilisateur peut se connecter on prend ce qu’il à saisit, on recalcule l’empreinte et elle doit correspondre à celle d’origine

Il n’est pas possible de récupérer un mot de passe perdu toutefois il est possible à un responsable AC Engineering d’en mettre un nouveau.

## Sécurité des échanges

Cette sécurité de mots de passe est toutefois mise à mal si l’utilisateur se fait écouter pendant qu’il envoie son mot de passe : c’est le principe de l’attaque dite « Man in the middle »

L’attaquant ne vas pas essayer de craquer le mot de passe, il écoute les échanges réseau de l’utilisateur (possible si en Wifi ou pire sur un réseau public type gare ,fast food, hotel …)

**SSL** + **certificat SSL**.

SSL va permettre de crypter la communication entre le client et l’applicatif la rendant inintelligible par un attaquant qui l’écoute. Cela permet de savoir que personne ne vous écoute.

Quand elle est activée vous avez un cadenas sur le navigateur 

SSL est toutefois insuffisant seul : en effet vous savez que personne ne vous écoute mais pas si vous ne parlez pas à l’attaquant. Même si vous échangez de manière sécurisée vous pouvez échanger avec le mauvais destinataire.

Pour cela on déploie un certificat SSL qui garantit l’identité du destinataire. Comment cela fonctionne – il ? Il y a un tiers de confiance, une société établie qui est certifiée par les navigateurs . C’est elle qui va donner le certificat et seulement au site authentique.

Quand on clique sur le cadenas on voit l’organisme de certification

Une image contenant texte, capture d’écran

Description générée automatiquement

Let’s encrypt est gratuit mais nécessite quelques manipulations, il y en a d’autres commerciaux : Thawte, Verisign

C’est la « carte d’identité » du site.

## Profils

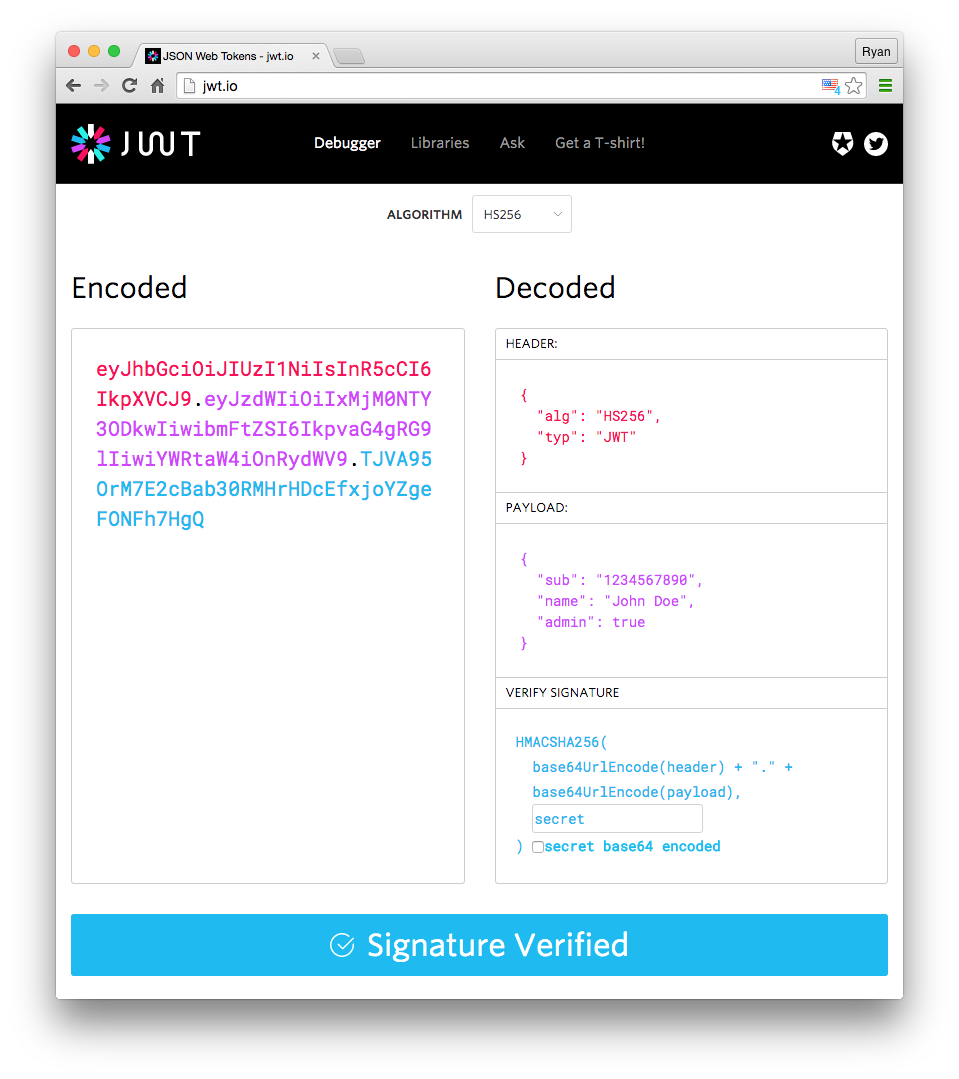
Dans l’application elle-même il faut ajouter une couche de sécurité :

* Sans être authentifié **la seule page possible est « login »**
* Une fois authentifié en tant que **client** par exemple « ACME » on ne voit que les données de « ACME »
  + Accès à la création d’une demande de patch manuelle ou automatique
  + Historique de ses tickets en cours ou traités
  + Changer son mot de passe
  + Chat avec AC Engineering
* Une fois authentifié en tant que « AC Engineering » vous avez accès à :
  + La liste des clients
  + Création d’un client
  + Ajout d’un utilisateur et changer son mot de passe (création ou si perdu)
  + (Dé-)Verrouiller un client (licence expirée / reconduite)
  + La corbeille des documents à patcher (tickets clients ouverts)
  + L’historique des traitements (tickets clients fermés)
  + Réponse au chat avec les clients

Il est évident qu’il ne faut pas qu’un client « malin » change son profil pour un autre client ou celui d’AC Engineering

Pour cela un jeton **JWT** est créé au login.

* Sans jeton il est systématiquement renvoyé au login
* Le jeton est attribué après un loin avec succès et contient son compte client et son profil
* Le jeton a une durée de vie max (ex 2h) au delà il est renouvelé ou déconnecté mais l’ancien jeton n’est plus valide ;
* Enfin ces jetons sont « signés » : toute modification le rend invalide



Le code bleu en bas à gauche est la signature (= somme de contrôle) du rouge et du violet qui est en clair. Si un client averti parvient quand même à modifier son profil (zone violette) ce qui est faisable, il ne peut pas régénérer la partie bleue correspondante. En effet elle est fabriquée par le serveur avec une clé secrète jamais retournée.

# Ecrans proposés (DRAFT)

Ceci est juste une base de discussion à affiner

## login

Page de login classique (très flexible mais doit être dans le thème retenu) sobre ou avec une image.

Cf exemples ci-dessus au début du document mais dans le thème retenu. S’il y a des images hors du logo : fournir les images choisies.

Une image contenant texte, capture d’écran

Description générée automatiquement

## Accueil

Une fois connecté : il y a souvent deux grandes variantes pour amener l’utilisateur vers ce qu’il veut faire.

**Option a)** Il y a la classique « barre de navigation » le plus souvent à gauche mais peut être en haut ou en bas. C’est le choix des applications classiques pro type « bureautique »

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

Avantages :

* Toujours accessible,
* Bonne visibilité de là ou on est et là où on peut aller
* Pratique à la souris et sur de bons écrans

Inconvénient :

* Prend un peu de place(Solution : une option peut prévoir un mode replié)

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

* Moins pratique en tactile ou sur petit écran : en mode replié les boutons peuvent être petits sur un écran tactile.

**Option b)**

Un look tablette avec des « tuiles de navigation »

Une image contenant capture d’écran, conception

Description générée automatiquement

Avantages :

* Laisse de la place pour des informations marketing type bandeau déroulant en haut, rapports et statistiques d’utilisation (versons ultérieure)
* Fonctionne même sur de petits écrans (les tuiles peuvent prendre ¼ ½ ou tout l’écran si petit)
* Pratique en tactile

Inconvénient :

* Une navigation supplémentaire (pour aller d’un groupe de fonctions à l’autre on repasse par la page des tuiles).

## Tableau historique

On affiche les tickets du client dans un tableau avec en colonne :

N° du ticket, date et heure de la demande, date et heure de traitement (ou « en cours » si pas fini), la plaque d’immatriculation du véhicule (ou la référence que le client a voulu mettre), le code du calculateur utilisé, les codes traitements demandés, le un lien vers le fichier d’origine à télécharger, un lien vers le fichier patché à télécharger

Ex (pas les bonnes colonnes mais pour illustrer)

Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

Filtrage :

Au dessus du tableau on lui propose de chercher

Par date (période début et optionnellement fin)

Par n° ticket

Par plaque d’immatriculation

Par code calculateur

En absence de filtre on voit les tickets du plus récent au plus ancien (sinon seuls ceux qui passent le filtre)

## Demande de traitement de fichier

A discuter je verrai bien en première à proche une page avec des zones dans l’ordre

1. **Soumettre votre fichier** à patcher (accepte fichier binaire ou zip)

Avec un bouton qui peut accepter du ‘drag and drop’ : exemples :

<https://primeng.org/fileupload>

1. **Choix sur liste du calculateur**
2. Soit un seul champ : on ouvre avec la liste et en cascade on choisit avec en option un filtre. Niveau 1 : maques, niveau 2 calculateurs

Une image contenant texte, capture d’écran, affichage, nombre

Description générée automatiquement

Un seul gadget, plutôt moderne, mais il faut une souris sinon peu pratique.

1. On sépare en deux champs : au-dessus on choisit la marque et en dessous

Contextuellement le calculateur (avec éventuellement un filtre)

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

1. **Codes des traitements à appliquer**
2. On peut mettre des cases à cliquer

Une image contenant capture d’écran, texte, Police, Graphique

Description générée automatiquement

Variante avec des boutons qui s’illuminent

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Variante avec un look type interrupteur

Une image contenant capture d’écran, texte, Police, Graphique

Description générée automatiquement

Il y a un bouton « Soumettre la demande »

* Vérifie que tout y est :
  + Un fichier
  + Un calculateur + une maque
  + Au moins un traitement à faire
* Si ce n’est pas le cas met en rouge ce qui manque + info en bulle d’information (Illustration)

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement

* Si c’est Ok on lui affiche une confirmation Oui / Annuler
* S’il confirme on lui affiche son ticket et on lui propose un autre traitement.

ex :

* Une image contenant texte, capture d’écran, logo, Police

  Description générée automatiquement

## Tableau des tickets à traiter

**AC Engineering uniquement.**

Similaire au tableau historique mais avec :

* Le code client en 1ère colonne
* Utile ? Le rappel de l’abonnement client : basique, gold, premium…
* Pas de colonne fichier patché
* N’y figurent que les tickets non traités du plus ancien au plus récent (à moins que l’abonnement premium fasse remonter en début de liste ?)
* Un bouton « Soumettre fichier patché » pour envoyer la réponse.

Si le fichier patché est soumis : le ticket disparait du tableau (et va dans l’historique)

Optionnel : enregistrer qui a envoyé le patch fichier (pour statistiques et rapports ultérieurs ou pour comprendre le contexte en chat)

## Liste des comptes clients et création

**AC Engineering uniquement.**

Tableau avec

Code client, Pays ?, niveau d’abonnement, date début abonnement, date fin abonnement, actif ou inactif, email de contact, nombre de tickets traités (0 au début), date du dernier ticket (vide au début) Illustration (on aura pas les photo par ex)

Une image contenant texte, logiciel, Logiciel multimédia, Icône d’ordinateur

Description générée automatiquement

Un bouton créer permet de créer un compte client.

Un bouton modifier permet de le modifier

## Liste des utilisateurs

**AC Engineering uniquement.**

Tableau avec

Code client, email, nom, prénom

Un bouton créer permet de créer un utilisateur

Un bouton modifier permet de le modifier

Un bouton changer le mot de passe permet de lui en affecter un nouveau (automatique ou saisi)

## Chat box

Solution tierce exemple

<https://www.tawk.to/software/live-chat>

Une image contenant texte, capture d’écran, Système d’exploitation, conception

Description générée automatiquement