

M4104C - Développement Application mobile



Dorian ABRAHAM - Alban DUCOS - Ali JAMALEDDINE
G5
Année Universitaire 2019-2020



Table des matières

Introduction	3
I - Éléments d'analyse	4
II - Nos choix techniques	13
A) Le QR code	13
B) La génération automatique du pdf	14
III - Description des fonctionnalités prévues	15
III - Difficultés rencontrées	17
IV - Liste des fonctionnalités réalisées	18
	22
Conclusions individuelles	28
Glossaire	31

Introduction

La vérification de la présence des élèves en cours est une tâche particulièrement redondante, non seulement pour ceux qui doivent signer la feuille, mais également pour le personnel enseignant qui doit vérifier s'il n'y a pas de falsification de signature ou bien si une personne présente n'a pas signé, et cela l'est encore plus pour la personne qui s'occupe de gérer les absences d'un département entier. Afin de pouvoir améliorer la gestion des absences à l'IUT, nous avons par conséquent décidé de développer une application Android qui permettrait de pouvoir faciliter l'ensemble des opérations relatives aux absences, à savoir justifier la présence d'un étudiant en cours et vérifier qu'un étudiant ne s'est pas fait passer pour un élève qui n'est pas en cours.

Comment, avec une application Android, pouvons-nous améliorer le système de gestion des absences ?

Dans un premier temps, nous allons parler des différents éléments d'analyse de notre application afin de pouvoir comprendre pourquoi nous avons procédé ainsi. Nous justifierons dans une deuxième partie nos choix techniques, puis dans une troisième partie, nous décrirons l'ensemble des fonctionnalités que nous avons imaginé possible d'intégrer dans notre application. Nous parlerons ensuite des difficultés que nous avons rencontrées au cours de ce projet et des solutions que nous trouvées afin de pouvoir les résoudre. Pour finir, nous ferons une liste des fonctionnalités que nous avons réussi à réaliser dans le temps imparti.

I - Éléments d'analyse

Nous avons choisi de développer une application qui permet à un étudiant, à l'aide d'un QR Code généré par l'enseignant, d'attester de sa présence dans un cours. L'application génère ensuite un fichier qui est envoyé par la suite au responsable des présences en cours.

La cible sera donc réduite aux membres de l'IUT souhaitant utiliser cette application. Soit plus précisément le professeur responsable du cours en question, les élèves assistant à ce cours ainsi que la personne de l'administration qui a en charge les absences des élèves (Madame Fleury pour notre cas).

L'objectif de notre application est le suivant: pouvoir permettre aux utilisateurs de fondamentalement faciliter et accélérer la tâche qu'est l'émargement. Les élèves pourront ainsi directement signaler leur présence ou leur absence, on pourra éviter alors tout oubli ou quiproquo.

D'autre part, cela évitera à l'enseignant d'appeler tout le monde ou dans le cas contraire faire passer la fiche d'émargement et risquer que quelqu'un soit oublié ou une fraude.

Enfin, le transfert direct de ses informations à la secrétaire lui facilitera la tâche de répertorisation sur l'ENT car des fichiers PDF lisibles lui seront transmis et la liste des absents aussi.

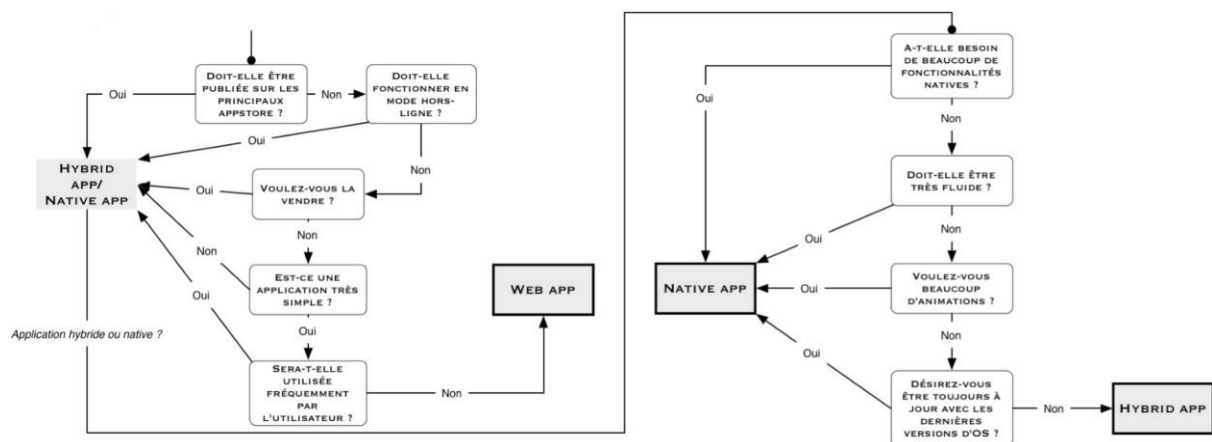
Avant de commencer le développement de cette application, il est important de déterminer quel type d'application nous nous apprêtons à développer avec le modèle mobile app/web app:

Portabilité	Application Web Navigateur Web
Développement/coût	Langages du web
Mise à jour	Possibilité de mises à jour quand des bugs sont fréquemment déclarés et pour maintenir une sécurité autour des documents.
Disponibilité	En ligne → Le professeur va sur un site internet qui génère des codes Qr, les élèves sortent leurs téléphones et scannent le code.
Fonctionnalités	<ul style="list-style-type: none"> - Générer un QR Code - A partir des étudiants qui ont scanné ce QR Code, générer un fichier PDF contenant la liste des étudiants présents en cours - Changement du code Qr toutes les 30 secondes

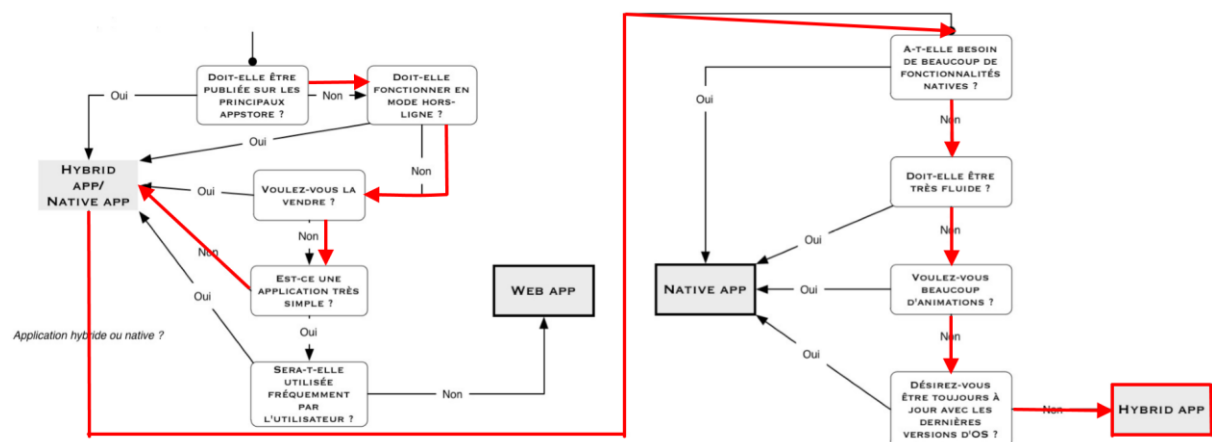
Nous optons donc pour le développement une application web.

Seulement, il ne s'agit pas du seul moyen de déterminer le type d'application que nous allons développer. En effet, il existe un autre modèle de développement, appelé le modèle de développement HybridApp qui nous permet de déterminer plus précisément le type d'application que nous allons développer par rapport à une suite logique de questions.

Nous allons maintenant déterminer le type de l'application que nous allons développer. Selon nos choix, notre application pourra être hybride, native ou bien Web. Afin de pouvoir déterminer le type de notre application, nous allons donc suivre le modèle de développement Hybrid App. Voici ci-dessous le modèle en question :



Et voici le chemin que nous avons emprunté :



Voici ci-dessous la justification à chacune des questions par lesquelles nous sommes passés:

- L'application doit-elle être publiée sur les principaux Appstore ?

Non. Cette application est conçue pour être utilisée uniquement par des personnes faisant partie d'un IUT, la publier sur les Appstore principaux n'est donc pas utile. Il s'agira donc seulement d'un apk qu'on pourra installer.

- Doit-elle fonctionner en mode hors-ligne ?

Non. Cette application devra envoyer directement au secrétariat de l'établissement et les élèves devront scanner un QR Code afin d'envoyer leur preuve de présence au professeur. Afin de réaliser cet ensemble d'opérations, un mode de fonctionnement hors ligne n'est pas nécessaire et pas envisageable.

- Voulez-vous la vendre ?

Cette application n'est pas destinée à la vente puisque nous allons la développer pour l'université de Limoges dans un premier temps. Elle en fera l'usage qu'elle juge nécessaire par la suite.

- Est-ce une application très simple ?

Cette application n'est pas une application très simple. En effet, il y a plusieurs fonctionnalités qui la rendent "complexe".

A partir de là, nous avons donc le choix entre une Hybrid App ou une Native App.

- A-t-elle besoin de beaucoup de fonctionnalités natives ?

Non. Cette application n'utilisera que le système de notifications présent sur la machine du professeur et du secrétariat.

- Doit-elle être très fluide ? / Voulez-vous beaucoup d'animations ?

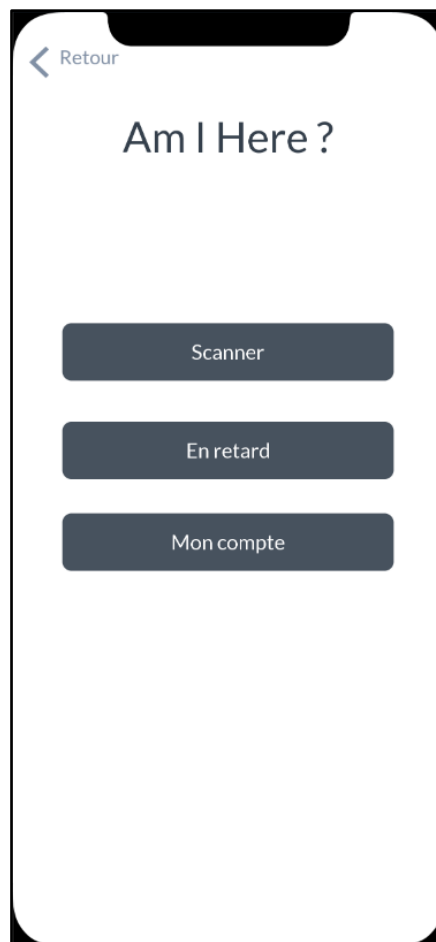
La fluidité dépend seulement de la qualité du serveur, plus elle est fluide le mieux c'est mais ce n'est pas une nécessité.

Cette application devra éviter un maximum de bugs mais la fluidité n'est pas une priorité pour le développement, tout comme les animations.

- Désirez-vous être toujours à jour avec les dernières versions d'OS ?

Ce n'est pas une nécessité puisque nous n'utilisons qu'une seule fonctionnalité d'OS et elle devra marcher sur tous les téléphones des étudiants qui ne sont pas forcément compatibles avec les dernières mises à jour d'OS.

Après avoir déterminé le type d'application que nous allons développer, qui est donc une Hybrid app, nous avons dû faire les maquettes de notre application afin d'avoir un rendu visuel du résultat final de notre application.



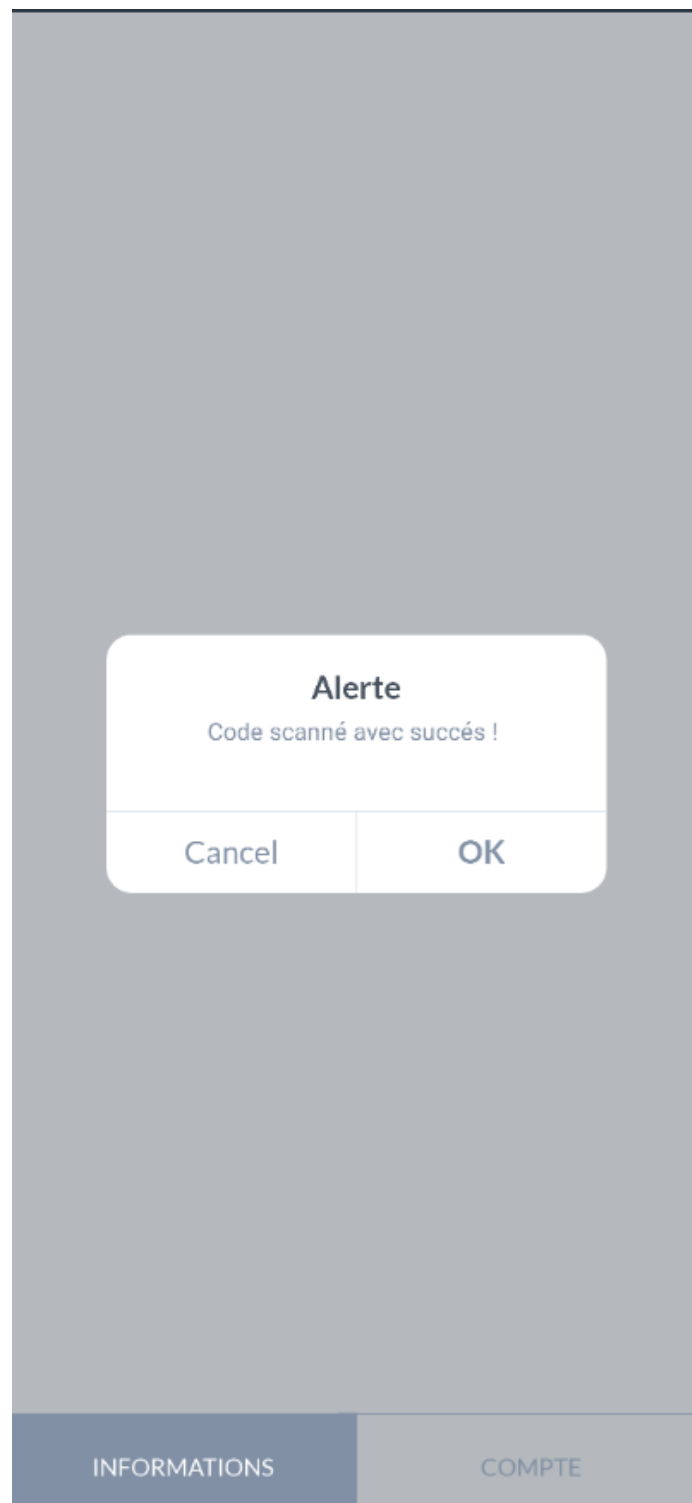
[Retour](#)

Am I Here ?

Se Connecter

Se connecter





Informations

Guide d'utilisation



II - Nos choix techniques

A) Le QR code

Un QR code est la représentation graphique d'un message. C'est une sorte de code barre en deux dimensions. Le-dit message peut contenir une URL, un e-mail, un numéro de téléphone, une vCard, etc. Dans la suite, nous nous intéresserons aux messages simples, les messages plus complexes n'étant que des extensions avec une syntaxe particulière.

La plupart des développeurs sont à l'aise avec les QR codes. Ils savent comment les scanner depuis leurs smartphones. De nombreuses "app" sont dédiées à cette fonctionnalité et on en possède souvent plusieurs. Les développeurs savent aussi comment créer des QR codes à l'aide des mêmes "app", des plugins Chrome ou Firefox, ou encore des sites web comme Unitag et SoQR.

Afin d'implémenter notre scanner de QR code, nous avons utilisé la bibliothèque Zxing. ZXing («zebra crossing») est une bibliothèque de traitement d'images de codes à barres implémentée en Java, avec des ports vers d'autres langages. Il prend en charge les codes-barres des produits 1D, industriels 1D et 2D.

ZXing est utilisé par la recherche sur le Web pour rendre indexables des millions de codes-barres sur le Web. Il constitue également la base de l'application Android Barcode Scanner et est intégré à Google Product and Book Search.

Dans notre application, nous avons dû implémenter un linear layout dans le fichier scanner.xml afin de définir une base, une structure pour le scanner. Ensuite, il nous a juste fallu implémenter la classe Scanner.java et le scanner est fonctionnel. L'utilisation de ZXing a été très bénéfique au groupe car elle nous a permis de ne pas perdre de temps sur ça et donc se concentrer sur d'autres fonctionnalités plus importantes.

B) La génération automatique du pdf

Pour ce qui est du fichier pdf, nous avons choisi d'utiliser Google Forms. En effet, cette fonction de google permet de faire des sondages en lignes et générer un fichier excel. Donc, chaque étudiant, lorsqu'il aura scanner le QR code, arrivera sur une page Google Forms où il remplira les informations demandées. Une fois ces informations récoltées pour tous les élèves, un fichier pdf sera envoyé par mail à l'administration.

Chaque Google Forms a un identifiant. L'élève, lors du scan du QR code, scannera donc l'identifiant du sondage. L'application elle prendra automatiquement l'identifiant et récupèrera la page. Une fois la page visitée, la réponse est envoyée automatiquement. Le google sheets est alors mis à jour, pour l'envoyer par mail on utilise google apps script, le google sheets est transformé par pdf, dans le script on peut indiquer les adresse mail auxquelles on veut envoyer.

III - Description des fonctionnalités prévues

L'ensemble des fonctionnalités que nous avons prévu est:

- ❖ Génération d'un code QR
- ❖ Scanner le code QR
- ❖ Génération d'un fichier PDF
- ❖ Transfert automatique fichier pdf
- ❖ Statistiques de présence / absence
- ❖ Consulter les élèves en retard

- **Génération d'un QR Code**

L'application sera en lien direct avec un générateur de QR code en ligne. Le professeur récupère le QR code proposé par l'application pour son cours et n'aura plus qu'à l'afficher devant ses élèves afin qu'ils puissent le scanner.

- **Scanner le Code QR**

Afin de scanner le QR code généré par l'application, celle-ci devra donc avoir accès à l'appareil photo des élèves. Ils devront donc autoriser l'accès à la caméra de leur appareil. Les élèves arrivent donc, après avoir scanner ce QR code sur une page avec leurs informations déjà pré-remplies grâce à leur informations présentes sur leur compte. Ils n'auront plus qu'à valider le formulaire et le professeur recevra les élèves présents sur sa page.

- **Génération d'un fichier PDF et transfert de ce fichier**

Une fois l'appel fait, l'application, qui est en lien avec le secrétariat de l'établissement, va générer un fichier pdf ayant pour contenu le nombre d'élèves présents et la liste des élèves absents avec Nom / Prénom / Promo / Groupe de TD.

- **Statistiques de présence / absence**

A partir de listes pré-établies, l'application effectue des calculs sur la base du nombre total d'élèves, ainsi, on peut construire des graphes et statistiques de présence/absence en fonction des itérations, c'est à dire au bout d'une semaine, mois, année.

Cette fonctionnalité n'est pas essentielle car les absences sont rares à l'IUT, mais les statistiques pourront par exemple être corrélées avec le taux de réussite, les difficultés rencontrées.

- **Consulter les élèves en retard**

L'enseignant a la possibilité de voir les élèves qui sont en retard, avec le temps qu'ils ont mis pour scanner le code QR. On compte tout élève qui n'a pas scanné le code QR dans les temps mais qui l'a fait par la suite comme élève en retard. L'application enverra une notification à l'enseignant pour chaque élève qui est arrivé en retard.

III - Difficultés rencontrées

La plus grosse difficulté que nous avons eue était la prise en main d'Android Studio. En effet, même si le langage utilisé est Java, coder une application en Android n'est pas du tout la même chose que faire un programme en Java, notamment avec la présence de fichiers XML, mais également au niveau des différentes classes à utiliser.

La seconde difficulté était le développement du scanner de QR Code. Tous les appareils Android n'ayant pas la même version, certains ne sont donc pas pour la plupart équipés de Google Lens ou d'un scanner de QR Code intégré. Il nous a donc fallu développer un scanner intégré dans l'application afin que tout le monde puisse s'en servir.

Une autre difficulté était le fait de réussir à prouver que la personne qui se disait présente avec l'application ne se faisait pas passer pour quelqu'un d'autre. Nous avons tout d'abord pensé à une connexion avec le compte unilim de la personne, mais cela s'est avéré ne pas être possible car nous n'avions pas accès à la base de données de l'IUT. Une solution a donc été de demander directement à l'utilisateur son nom, mais également son numéro étudiant. Une fois la page du formulaire visitée, la réponse est envoyée automatiquement ce qui permet d'éviter toute usurpation d'identité.

Un des derniers problèmes que nous avons rencontrés est l'envoi par mail du fichier PDF. Afin d'y pallier, nous avons décidé d'utiliser Google App Scripts, qui nous a grandement facilité la tâche. En effet, à l'aide d'un script, on peut indiquer les adresses mail auxquelles on veut envoyer le fichier, et il est également possible de créer un déclencheur qui envoie automatiquement le fichier à une heure donnée. Nous avons pour l'instant configuré l'envoi du mail de manière à ce que celui-ci se fasse quotidiennement.

IV - Liste des fonctionnalités réalisées

Voici ci-dessous la liste des fonctionnalités que nous avons réussi à réaliser dans les temps:

- La génération d'un QR Code;
- Ajouter la possibilité de scanner le QR Code directement depuis l'application;
- Générer un fichier PDF à partir des réponses du formulaire;
- Génération d'une partie explicative de l'application
- Transférer automatiquement par mail le fichier PDF à la secrétaire.

Il est important de noter que nous nous sommes écartés du chemin que nous avons choisi au départ. En effet, notre application devait permettre de consulter les élèves en retard, mais également de faire des statistiques sur le taux de présence et d'absence des étudiants. Comme le Google Sheets créé à partir des résultats du formulaire permet de savoir à quelle heure un élève y a répondu, la fonctionnalité concernant les élèves en retard n'était donc pas nécessaire. En ce qui concerne la génération de statistiques, nous n'avons au final pas jugé nécessaire de le faire car trop complexe et avons préféré nous focaliser sur les fonctionnalités clés de notre application, à savoir le scanner de QR Code et l'envoi automatique du fichier PDF par mail.

La génération du code QR :

Voici les différentes étapes suivies :

- Lien de création du google forms :

https://docs.google.com/forms/d/1EiTGHo2IRGvbl12-bQVaSs_yHA1I5oZxs1eqP0yVdig/edit

- On obtient le lien du prefill :

https://docs.google.com/forms/d/e/1FAIpQLScm3SKnOnOKBFsyx23_jFZfCHXw1UWyM6wdLOyKhdmhRofg/viewform?usp=pp_url&entry.1004622673=00&entry.1827464113=JAMALEDDINE

- On remplace le viewform par un formResponse pour directement enregistrer la réponse dès qu'on a visité le lien.

- On obtient alors:

https://docs.google.com/forms/d/e/1FAIpQLScm3SKnOnOKBFsyx23_jFZfCHXw1UWyM6wdLO-yKhhdmhRofg/formResponse?usp=pp_url&entry.1004622673=00&entry.1827464113=JAMALEDDINE

- On retire les champs pour avoir le lien personnalisable

https://docs.google.com/forms/d/e/1FAIpQLScm3SKnOnOKBFsyx23_jFZfCHXw1UWyM6wdLO-yKhhdmhRofg/formResponse?usp=pp_url&entry.1004622673=&entry.1827464113=

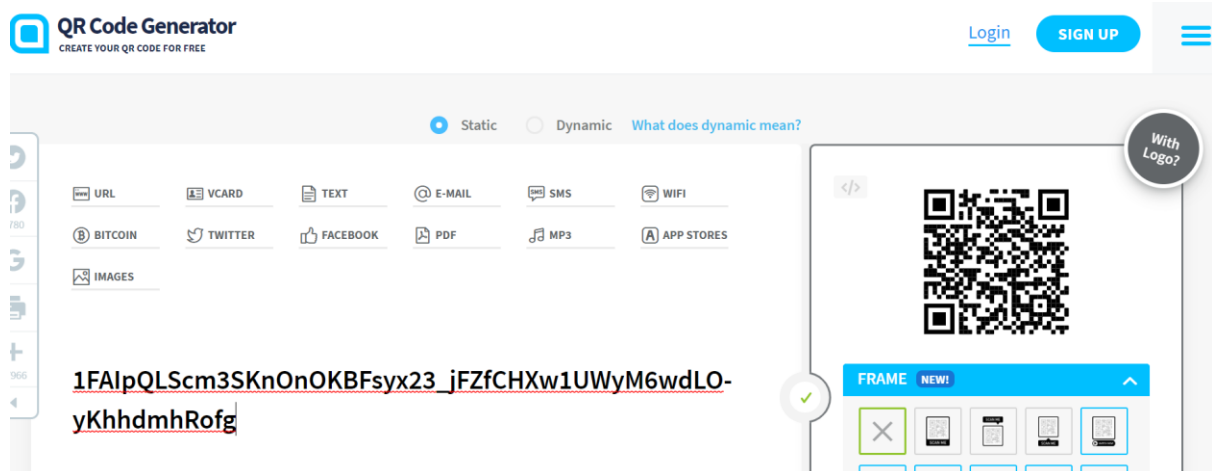
- De cette façon on peut utiliser ce lien pour directement enregistrer les étudiants,
On a juste besoin de l'identifiant du google forms

C'est à dire :

[1FAIpQLScm3SKnOnOKBFsyx23_jFZfCHXw1UWyM6wdLO-yKhhdmhRofg](https://docs.google.com/forms/d/e/1FAIpQLScm3SKnOnOKBFsyx23_jFZfCHXw1UWyM6wdLO-yKhhdmhRofg)

On génère alors un code QR à partir de cet identifiant, ce sera le code QR que l'étudiant va scanner.

Cette opération est visible sur le code, dans la classe Scanner.



Voici le code :



Donc, si jamais le professeur veut changer de forms, il n'aura qu'à récupérer l'identifiant et générer le code QR.

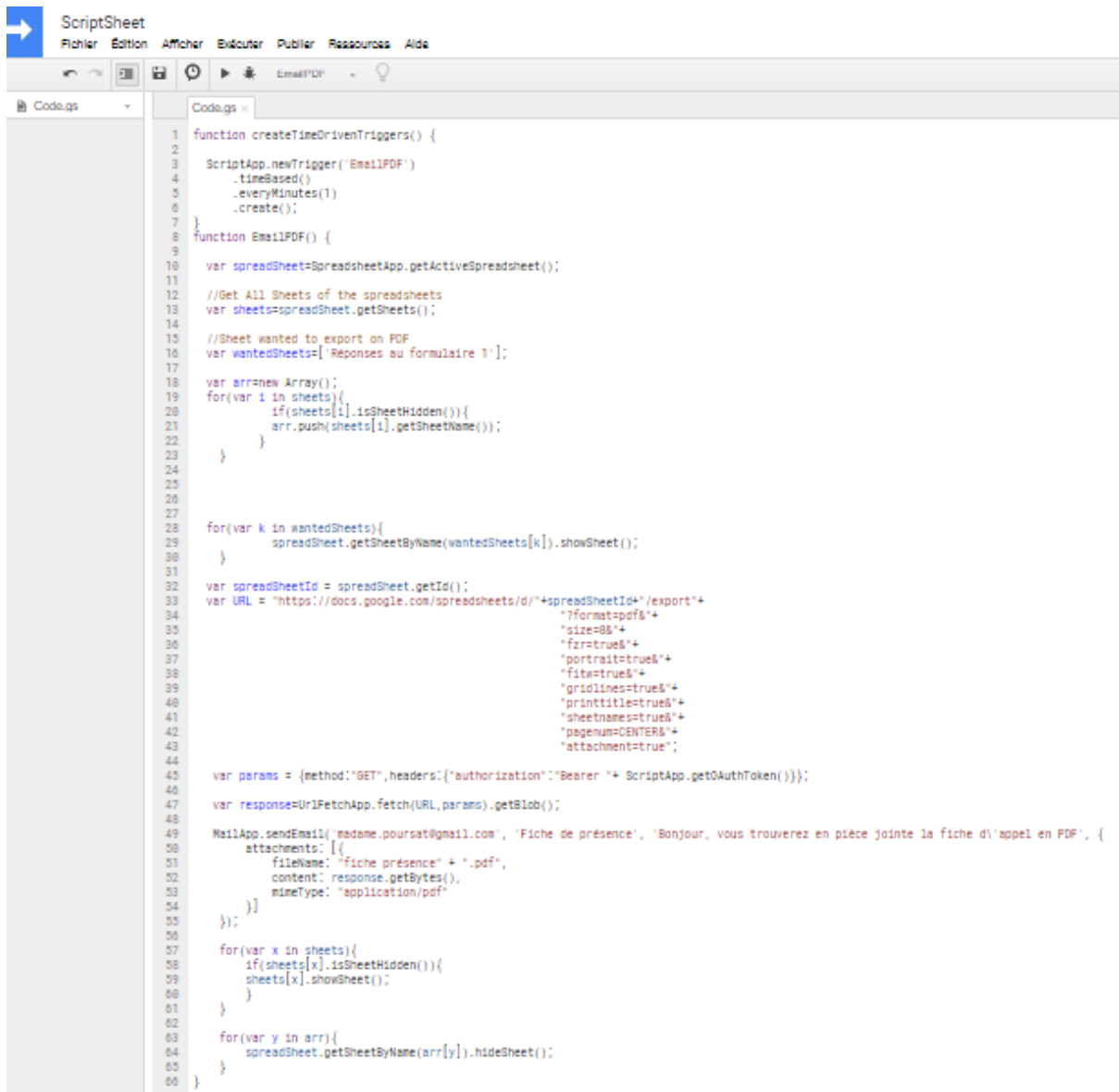
Génération du fichier PDF:

En créant le Forms, il faut aussi créer le Google Sheets qui va avec, ce dernier est automatiquement mis à jour à chaque réponse, en revanche la conversion en PDF et l'envoi par mail n'est pas automatique, nous avons utilisé alors Google App Scripts.

Google Apps Script est un langage de programmation propre à Google. Il permet de réaliser des scripts à l'intérieur de certains services de Google, dont Google Apps¹. Ainsi il est possible de modifier le comportement du logiciel tableur, Google Spreadsheets, à l'instar de VBA pour MicrosoftExcel.

Le script est disponible sur le Drive et peut être manuellement exécuté pour tester, nous avons également placé un trigger qui se lance quotidiennement et envoie le PDF à l'adresse mail fournie.

Interface google app script :



```
1 function createTimeDrivenTriggers() {
2
3   ScriptApp.newTrigger('EmailPDF')
4     .timeBased()
5     .everyMinutes(1)
6     .create();
7 }
8 function EmailPDF() {
9
10  var spreadsheet=SpreadsheetApp.getActiveSpreadsheet();
11
12  //Get All Sheets of the spreadsheets
13  var sheets=spreadsheet.getSheets();
14
15  //Sheet wanted to export on PDF
16  var wantedSheets=["Réponses au formulaire 1"];
17
18  var arr=new Array();
19  for(var i in sheets){
20    if(sheets[i].isSheetHidden()){
21      arr.push(sheets[i].getSheetName());
22    }
23  }
24
25
26
27
28  for(var k in wantedSheets){
29    spreadsheet.getSheetByName(wantedSheets[k]).showSheet();
30  }
31
32  var spreadsheetId = spreadsheet.getId();
33  var URL = "https://docs.google.com/spreadsheets/d/"+spreadsheetId+"/export"+
34    "?format=pdf&"+
35    "size=8&"+
36    "fz=true&"+
37    "portrait=true&"+
38    "fitw=true&"+
39    "gridlines=true&"+
40    "printtitle=true&"+
41    "sheetnames=true&"+
42    "pagenum=CENTER&"+
43    "attachment=true";
44
45  var params = {method:'GET',headers:{"authorization":"Bearer "+ ScriptApp.getOAuthToken()}};
46
47  var response=UrlFetchApp.fetch(URL,params).getBlob();
48
49  MailApp.sendEmail('madame.poursat@gmail.com', 'Fiche de présence', 'Bonjour, vous trouverez en pièce jointe la fiche d'appel en PDF', {
50    attachments: [{
51      fileName: "fiche présence" + ".pdf",
52      content: response.getBytes(),
53      mimeType: "application/pdf"
54    }]
55  });
56
57  for(var x in sheets){
58    if(sheets[x].isSheetHidden()){
59      sheets[x].showSheet();
60    }
61  }
62
63  for(var y in arr){
64    spreadsheet.getSheetByName(arr[y]).hideSheet();
65  }
66 }
```

Code :

```
function createTimeDrivenTriggers() {  
    ScriptApp.newTrigger('EmailPDF')  
        .timeBased()  
        .everyMinutes(1)  
        .create();  
}  
function EmailPDF() {  
    var spreadsheet=SpreadsheetApp.getActiveSpreadsheet();  
  
    //Get All Sheets of the spreadsheets  
    var sheets=spreadsheet.getSheets();  
  
    //Sheet wanted to export on PDF  
    var wantedSheets=['Réponses au formulaire 1'];  
  
    var arr=new Array();  
    for(var i in sheets){  
        if(sheets[i].isSheetHidden()){  
            arr.push(sheets[i].getSheetName());  
        }  
    }  
  
    for(var k in wantedSheets){  
        spreadsheet.getSheetByName(wantedSheets[k]).showSheet();  
    }  
  
    var spreadsheetId = spreadsheet.getId();  
    var URL = "https://docs.google.com/spreadsheets/d/"+spreadsheetId+"/export"+  
        "?format=pdf&"+  
        "size=0&"+  
        "size=0&"+  
        "fzr=true&"+  
        "portrait=true&"+  
        "fitw=true&"+  
        "gridlines=true&"+  
        "printtitle=true&"+  
        "sheetnames=true&"+  
        "pagenum=CENTER&"+  
        "attachment=true";  
  
    var params = {method:"GET",headers:{"authorization":"Bearer "+ ScriptApp.getOAuthToken()}};  
    var response=UrlFetchApp.fetch(URL,params).getBlob();  
  
    MailApp.sendEmail('madame.poursat@gmail.com', 'Fiche de présence', 'Bonjour, vous trouverez en pièce jointe la fiche d\'appel en PDF', {  
        attachments: [{  
            fileName: "fiche présence" + ".pdf",  
            content: response.getBytes(),  
            mimeType: "application/pdf"  
        }]  
    });  
  
    for(var x in sheets){  
        if(sheets[x].isSheetHidden()){  
            sheets[x].showSheet();  
        }  
    }  
  
    for(var y in arr){  
        spreadsheet.getSheetByName(arr[y]).hideSheet();  
    }  
}
```

Le script est aussi disponible sur le Drive fourni.

Le trigger peut être programmé (voir script) ou créé à partir de l'interface proposée par Google.

Ajouter un déclencheur pour ScriptSheet

Choisir la fonction à exécuter

EmailPDF

Choisissez le déploiement à exécuter

Head

Sélectionnez la source de l'événement

Déclencheur horaire

Sélectionnez le type de déclencheur temporel

Quotidien

Sélectionnez une heure de la journée

Paramètres du déclencheur

Re

Le trigger quotidien :

Google Apps Script

Rechercher un nom de projet

M

Nouveau projet

Projets suivis

Mes projets

Tous les projets

Partagés avec moi

Corbeille

Mes exécutions

Mes déclencheurs

Premiers pas

Paramètres

État du service

Aide

< ScriptSheet > Déclencheurs

1 déclencheur déclencheurs affichés

EFFACER LES FILTRES

Propriétaire: Moi

Propriétaire	Dernière exécution	Déploiement	Événement	Fonction	Taux d'erreur
Moi	-	Head	Basé sur l'heure	EmailPDF	-

+ Ajouter un déclencheur

Réception du mail :

Fiche de présence

Boîte de réception x



madame.poursat@gmail.com

20:05 (il y a 27 minutes)



À moi ▾

Bonjour, vous trouverez en pièce jointe la fiche d'appel en PDF



L'objet et le corps du message peuvent bien évidemment être modifiés sur le script.

Le pdf reçu :

PDF fiche présence.pdf

Rechercher dans les messages

Ouvrir avec Google Docs

Nouveau message

Boîte de réception 2

Messages suivis

En attente

Fiche d'appel n°1 (réponses)

Réponses au formulaire 1

Horodateur	N° INE	Nom
27/03/2020 19:33:06	00	Jamaledine
27/03/2020 19:33:14	01	DUCOS
27/03/2020 19:33:25	03	Abraham
27/03/2020 19:34:11	04	DUPONT

Du point de vue utilisateur ça se passe comme suit :

AmlHere

Am I Here ?

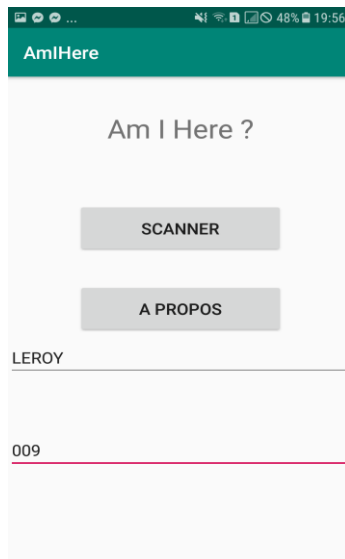
SCANNER

A PROPOS

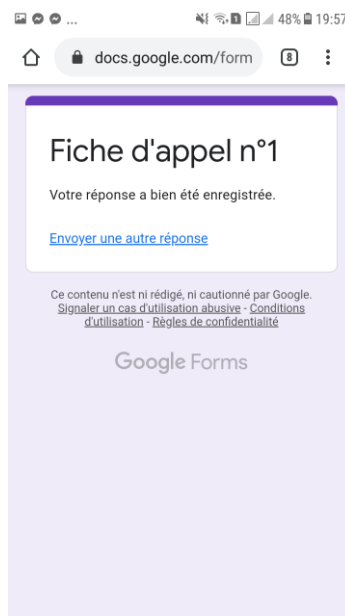
nom

numéro étudiant

1. L'utilisateur renseigne son nom et son Numéro



2. Il clique sur scanner



3. Il est automatiquement redirigé vers le forms ou la réponse est automatiquement enregistrée.

La question suivante se pose, pourquoi ne pas directement utiliser le lien du forms ?

Voici la structure générale du lien :

[https://docs.google.com/forms/d/e/\(identifiant\)/\(typeduForm\)?usp=pp_url&entry.1004622673](https://docs.google.com/forms/d/e/(identifiant)/(typeduForm)?usp=pp_url&entry.1004622673)
=(entrée)

Nous récupérons l'identifiant et les entries (la suite de numéros en rouge), et remplaçons l'entrée par le contenu du formulaire dans l'appli.

L'idée principale était de faciliter la tâche aux professeurs et mettre un custom url ou rediriger rapidement la page pour que le lien ne soit pas partagé entre les élèves. Nous avons malheureusement pas eu le temps d'implémenter cette fonctionnalité, l'application n'est utilisable qu'avec le forms fourni car nous n'avons pas pu programmer la récupération de l'entry, ([1004622673](#)) dans l'exemple.

Le code qu'on prévoyait d'utiliser :

```
<iframe  
src="https://docs.google.com/forms/d/e/1FAIpQLScwDvYB_AhTxlldCHrjxfe001J6XLm31gj  
bcf3mjyqyYxzXiQ/viewform?embedded=true" width="640" height="510" frameborder="0"  
marginheight="0" marginwidth="0">Chargement...</iframe>
```

On allait incorporer grace au iframe le forms dans un site et utiliser un script pour masquer l'url.

Voici un exemple d'un script :

```
<script type="text/javascript"> $(document).ready(function(e) {  
$('iframe').attr('src','http://www.flickr.com/'); }); </script> <body>  
<iframe src="" /> </body>
```

On pourrait aussi empêcher le clic droit :

```
<script language="JavaScript"> <!-- var message="Your message goes  
here."; function click(e) { if (document.all) { if (event.button == 2) {  
alert(message); return false; } } if (document.layers) { if (e.which ==  
3) { alert(message); return false; } } } if (document.layers) {  
document.captureEvents(Event.MOUSEDOWN); } document.onmousedown=click; //  
-->
```

Accès au formulaire, Script et sheet :

Nous avons créé une adresse mail pour que vous puissiez y accéder.

Gmail : madame.poursat@gmail.com

Mot de passe : madamepoursat87

Conclusions individuelles

Dorian:

Même si j'étais au départ sceptique concernant notre idée de départ, je suis malgré tout convaincu du résultat final de notre application. Cependant, je trouve ça dommage de ne pas avoir pu utiliser la base de données de l'université pour pouvoir prouver notre absence. Même si l'application n'est pas censée être belle à voir, je trouve cela dommage de ne pas avoir intégré certaines fonctions comme par exemple un mode sombre ou bien d'avoir affiné un peu l'interface graphique afin de rendre l'application plus attrayante.

Malgré les craintes que j'avais vis-à-vis d'Android Studio, celui-ci s'est révélé assez pratique à prendre en main grâce aux cours et aux TP que nous avons eu. De plus, le fait de pouvoir tester directement l'application que l'on est en train de coder sur son téléphone est un grand plus, car cela nous permet de tester son ergonomie et d'avoir un vrai aperçu de ce que donne l'application.

Personnellement, je ne pensais pas qu'intégrer un scanner de QR Code dans notre application était une bonne idée car pour moi, la plupart des téléphones l'avaient déjà d'intégré et on avait juste à ouvrir l'application qui permettait de scanner les QR Codes (dans la majorité des cas, Google Lens). Cependant, l'avoir fait est une bonne chose car cela nous a permis d'en apprendre plus sur le développement d'applications Android.

Alban :

Le développement de cette application fût ma première vraie expérience dans ce domaine et m'a aidé à me confirmer dans mes choix de poursuite d'études. En effet, j'ai toujours été attiré par le développement d'applications mais je n'avais encore rien eu de concret, cette matière a donc été une confirmation d'orientation pour l'année prochaine.

De plus, ce projet a été une bonne expérience de développement notamment dû à l'utilisation d'un nouveau logiciel qui ne nous a pas laissé sans craintes, Android Studio.

La création d'une application est une méthode de travail nouvelle. Même si cette application peut paraître simple, c'est la première fois que je travaille sur ce type de projet. Nous sommes passé d'une idée à imaginer le fonctionnement et enfin au codage de cette application.

Le fait d'organiser des séances de TP pour se familiariser avec Android Studio en parallèle avec les séances de TD qui sont exclusivement consacrées au développement de notre application.

Nous avons été également très accompagnés par notre professeur qui nous a beaucoup aiguillé sur les choix d'outils à utiliser dans notre application.

Si je devais refaire ce projet, il y a un point sur lequel je serai plus exigeant c'est le design général de l'application. En effet, le manque de temps et de compétences dans ce domaine ne nous a pas laissé le temps de nous pencher en détail sur le design global.

De plus, si je devais refaire ce projet avec plus de temps, j'essaierais d'établir un système d'identification. Cette mise à jour entraînerait l'élaboration d'une base de données. C'est cet aspect de la programmation mobile que j'aurai souhaité découvrir plus en détail.

Ali :

Les applications mobiles font désormais partie intégrante de la stratégie numérique et la demande d'applications mobiles augmente rapidement dans le monde entier. Nous pouvons trouver des millions d'applications professionnelles, d'applications sur le mode de vie, d'applications éducatives, d'applications à la demande, etc. sur Google Play Store et l'App Store. La technologie dans le domaine du développement d'applications mobiles progresse de jour en jour.

Ce projet était une première introduction au développement mobile pour moi. Il a été difficile de démarrer : incertitude, inconnus, choses auxquelles il faut réfléchir et décisions à prendre. Aucune idée par quoi commencer et où. Mais je ne me suis pas découragé. Quand j'avais besoin de quelque chose, je le cherchais sur Google. En général, il y avait des réponses utiles, des extraits de code ou des tutoriels vidéo pour presque tout. Lentement, j'ai commencé à me sentir beaucoup plus à l'aise pour basculer vers des choses inconnues.

J'ai enfin commencé à apprendre comment les application qu'on utilise chaque jour fonctionnent !

J'ai pu a travers du module et de ce projet de me projeter et d'imaginer mes propres applications, il me reste cependant beaucoup de choses à apprendre mais je me réjouis tout de même de cette expérience car elle m'a permis de découvrir le monde du développement mobile, qui n'est pas forcément facile d'accès.

J'ai rencontré d'innombrables difficultés, j'ai pu néanmoins y remédier grâce aux forums en ligne qui sont très instructifs, j'ai aussi pu utiliser du javascript, ce que je n'aurais pas pu imaginer, cela ma permis de comprendre qu'il y a une dépendance entre ce qui est mobile et ce qui est web.

Je regrette le fait de ne pas avoir pu incorporer d'autres fonctionnalités telles que la connexion et l'enregistrement des élèves dans une base de données, et aussi la sécurisation.

Nous avons beaucoup d'idées mais nous avons aucune idée de comment les implémenter, cela m'a permis de comprendre qu'il y a un vrai travail de réflexion derrière tout projet de développement.

Il ne s'agit pas simplement d'écrire des activités, cliquer sur un bouton et basculer vers une autre activité, il faut aussi penser à la persistance des données, par exemple quand on oriente l'écran, les données dans les formulaires ne disparaissent pas. Ou faire basculer des données de formulaire d'une activité à une autre. Il faut aussi penser à l'expérience utilisateur aussi par exemple.

Ceci fut un expérience très agréable pour moi et je prévois de développer un jeu que j'ai imaginé par la suite.

Glossaire

Android : Android est le système d'exploitation mobile créé par Google. Il équipe la majorité des téléphones portables du moment (smartphones)

APK : Un APK est une collection de fichiers compressée pour Android. L'ensemble constitue un « paquet ». Un APK est similaire à un paquet deb ou RPM.

Application Hybrid : Une application hybride est une application utilisant le navigateur web intégré du support et les technologies Web pour fonctionner sur différents OS.

Application Native : Une application native est une application mobile qui est développée spécifiquement pour un des systèmes d'exploitation utilisé par les smartphones et tablettes (iOS, Android, etc.).

Application Web : une application web est une application manipulable directement en ligne grâce à un navigateur web et qui ne nécessite donc pas d'installation sur les machines clientes, contrairement aux applications mobiles

ENT : Abréviation de Espace Numérique de Travail, il permet aux élèves et aux enseignants de consulter et d'échanger des contenus

Google Lens : Google Lens est une technologie de reconnaissance d'image développée par Google, conçue pour afficher des informations pertinentes relatives aux objets qu'il identifie à l'aide d'une analyse visuelle basée sur un réseau de neurones

Java : Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy, présenté officiellement le 23 mai 1995 au SunWorld.

Linear layout : LinearLayout est un layout simple à utiliser, qui va vous permettre de mettre les différents éléments, donc un élément c'est par exemple un texte ou un bouton, les uns à la suite des autres.

OS : En informatique, un système d'exploitation est un ensemble de programmes qui dirige l'utilisation des ressources d'un ordinateur par des logiciels applicatifs.

PDF : Le Portable Document Format, communément abrégé en PDF, est un langage de description de page présenté par la société Adobe Systems en 1992 et qui est devenu une norme ISO en 2008.

XML files : XML Schema publié comme recommandation par le W3C en mai 2001 est un langage de description de format de document XML permettant de définir la structure et le type de contenu d'un document XML