

SYSTÈMES DE GESTION DE BASES DE DONNÉES

Site d'un opérateur téléphonique

Janvier
22
2017

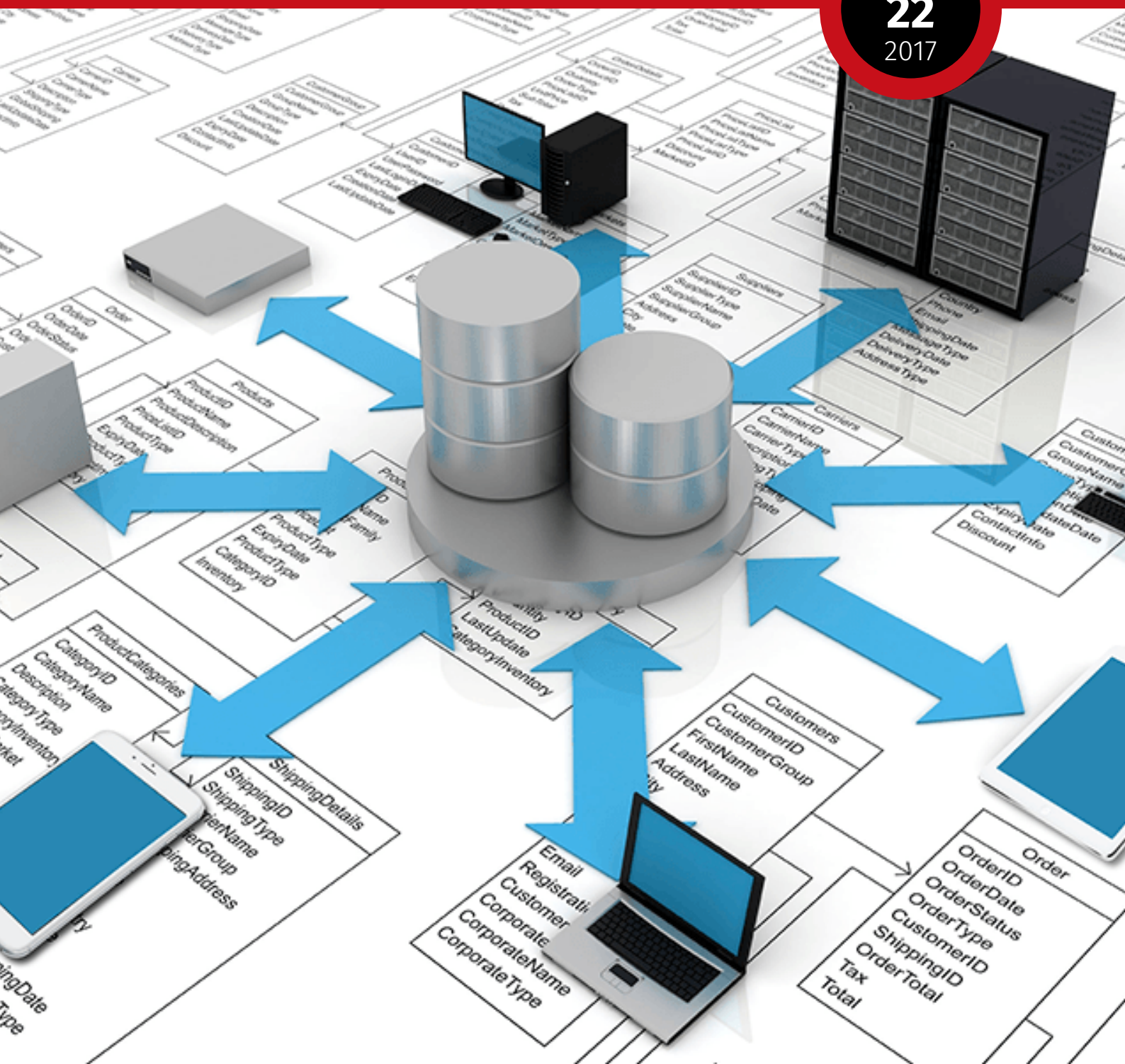


Table des matières

I	Introduction	4
1	• Travail réalisé	5
1.1	Sujet traité.	5
1.2	Détail des travaux	5
1.3	Installation de l'application	5
1.3.1	Installation de la base de données.	6
1.3.2	Installation du code source	6
1.3.3	Test de bon fonctionnement	7
2	• Élaboration de la base de données.	8
2.1	Réalisation de l'analyse Entité-Relation.	8
2.1.1	Entités.	8
2.1.2	Les Relations	9
2.1.3	Spécialisation	10
2.1.4	Diagramme récapitulatif.	10
2.2	Définition des tables à utiliser	11
2.2.1	Gestion des relations N..N	11
2.2.2	Gestion des relations 1..N	11
2.2.3	Gestion de la généralisation isa	11
2.2.4	Diagramme final	11
3	• Plateforme réalisée.	13
3.1	Les différentes vues	13
3.1.1	Liste des vues	13
3.1.2	Navigation dans l'application	13
3.2	Gestion des utilisateurs.	13
3.2.1	Les différents types d'utilisateurs.	13
3.2.2	Connexion et déconnexion	14
3.2.3	Édition du profil en tant que client.	15
3.2.4	Édition des profils en tant qu'administrateur.	15
3.3	Les téléphones	15
3.3.1	Consulter la liste des téléphones	16
3.3.2	Créer et modifier un téléphone	16
3.3.3	Acheter un téléphone	17
3.4	Consultation des factures	17
3.4.1	Liste des factures.	17
3.4.2	Détail des consommations	18
3.5	Gestion des abonnements	18
3.5.1	La liste des formules.	18
3.5.2	Consultation des détails d'une formule	20
3.5.3	Consultation des forfaits étrangers.	21
3.5.4	Création, édition et suppression d'une formule ou d'un forfait étranger.	21
3.5.5	Souscription à un forfait et consultation des forfaits.	22
4	• Aspects techniques de la plateforme.	24
4.1	Architecture de la plateforme	24
4.1.1	Architecture MVC	24
4.1.2	Appel des vues	24
4.1.3	Les fichiers affichés	25

4.1.4	Scripts, styles et autres éléments	25
4.2	Connexion à la base de données	25
4.2.1	Connexion et déconnexion	25
4.2.2	Exécution de requête	25
4.2.3	Traitement des chaînes de caractères	26
4.2.4	Utilisation de fonctions	26
4.3	Les requêtes SQL	26
4.3.1	Insertion dans la base	26
4.3.2	Une requête plus évoluée	27
4.4	La génération du PDF de consommation détaillée	29
4.4.1	Récupérer la consommation dans la base de données	29
4.4.2	Génération du fichier XML	29
4.4.3	La transformation en PDF	30
5	• Améliorations possibles	31
	Conclusion	32

Liste des figures

1	Écran d'accueil de l'application	7
2	Le diagramme Entité-Relation de notre base de données	10
3	Relations entre les tables de la base de données	12
4	Panneau de navigation pour l'application	14
5	Les fenêtres de connexion et déconnexion des utilisateurs	14
6	Gestion du compte par un client	15
7	Gestion des comptes clients par un administrateur	16
8	Consulter la liste des téléphones	16
9	Affichage des détails d'un téléphone	17
10	Création d'un téléphone	18
11	Confirmation de l'achat d'un téléphone	18
12	Liste des factures	19
13	Listes générées dans le PDF	19
14	Vue présentant la liste des formules de Centrale-Télécom	19
15	Détails d'une formule	20
16	Consultation d'éléments associés à une formule	20
17	Filtrage des formules avec l'outil de recherche	21
18	Consultation de la liste des forfaits étrangers et de leurs détails	21
19	Modale de création d'un nouvel abonnement	22
20	Modale pour associer des éléments à une formule	22
21	Vue résumant tous les achats (formules et téléphones) d'un client	23

Liste des tables

1	Valeurs stockées dans \$_SESSION pour les différents profils d'utilisateurs	14
---	---	----

Introduction

Ce rapport détaille l'ensemble de nos travaux réalisés pour le module de Système de Gestion de Base de données. Le projet utilisé pour manipuler les concepts vus en cours est celui relatif à la téléphonie, qui vous est rappelé dans la première partie de ce document.

Afin de présenter notre travail, nous vous précisons dans un premier temps le contexte général d'étude. Ensuite, nous nous intéresserons plus en détail à l'élaboration de la base de données, en montrant en particulier le passage du modèle entité-association à celui des tables utilisées dans la pratique.

Dans un troisième temps, nous détaillerons les fonctionnalités mises en place dans la plateforme et expliquerons le fonctionnement de cette dernière. Enfin, nous préciserons les réalisations techniques principales, que ce soit concernant la base de données, l'architecture de l'application ou encore la génération automatique de PDF.

Enfin, avant de conclure sur ce travail nous préciserons quelques axes d'amélioration possibles pour cette application, mais qui n'ont pas pu être réalisés, car plus gourmands en temps.

En cas de questions sur le contenu de ce rapport, le code ou l'installation de notre application, n'hésitez pas à nous contacter aux adresses suivantes :

`damien.douteaux@ec113.ec-lyon.fr`

`valentin.demeusy@ec113.ec-lyon.fr`

De plus, vous pourrez retrouver l'intégralité de notre code et ce rapport sur le répertoire GitHub suivant :

`https://github.com/stity/projet_sql`

Remarque importante : Ce rapport constitue la version définitive de celui qui vous a été remis dans la première archive. En particulier, la procédure d'installation, les aspects techniques et les schéma Entité-Relation et de base de données ont été fortement mis à jour.

1 • Travail réalisé

1.1 | Sujet traité

Le sujet que nous avons choisi de traiter est celui du site d'une compagnie de téléphone mobile. Ce dernier vous est rappelé ci-dessous :

« La compagnie de téléphonie mobile Centrale-Telecom en concurrence directe avec les services Orange, SFR et BouyguesTelecom, offre différents types de forfaits au public. Chaque formule est proposée à un certain prix et varie en nombre d'heures de communication par mois et en options. Les options concernent :

- ◉ Un certain nombre de SMS par mois,
- ◉ L'accès internet,
- ◉ Un volume de données transférées par internet(en ko),
- ◉ Des plages horaires d'appels illimités,
- ◉ Plus toutes les options que vous voudrez ajouter pour écraser la concurrence.

Par ailleurs une offre de téléphones existe, chaque téléphone étant d'une part caractérisé par ses caractéristiques (écran, capacités internet, TV, appareil photo, video numérique, mémoire,...) et d'autre part peut être éventuellement associé à une offre à un prix promotionnel.

Pour ce qui est de la facturation, on gèrera les communications selon la tarification par forfait (locale, nationale et internationale par pays) sachant qu'un forfait comprend un certain montant mensuel en communications. On n'oubliera pas les communications hors forfait. On gèrera également la facturation des autres services (SMS, Internet, MMS). Enfin la facturation sera associée à l'éventuel règlement ou non de la facture ainsi que le blocage éventuel d'une ligne en cas d'impayés.

Le travail à réaliser consiste à concevoir une base de données gérant les différents forfaits et la clientèle, ainsi qu'offrir des informations sur les forfaits, les promotions du mois et éventuellement consultation de l'état de la consommation des clients. ».

1.2 | Détail des travaux

Pour traiter ce sujet, nous avons alors réalisé les éléments suivants, qui vous seront présentés tout au long de ce rapport :

- ◉ Conception du modèle de base de données en partant d'un modèle entité-association. Nous vous présenterons ainsi la version entité-association, et les réflexions qui nous ont conduite au modèle de base de données associé.
- ◉ Implémentation de la base de données avec MySQL. Cette implémentation ne sera pas détaillée, cependant nous précisons à la section suivante comment installer notre base de données (et remplir ses tables automatiquement) sur votre poste.
- ◉ Réalisation d'une application utilisant cette base de données. L'application utilise du PHP en langage côté serveur pour interroger la base de données. Une utilisation de XML et d'une feuille XSL a également été réalisée dans cette application.

Les détails des différentes parties sont exposés dans la suite de ce rapport.

1.3 | Installation de l'application

Avant de présenter cette application, nous allons détailler la manière d'installer cette application sur un poste. On supposera dans cette démarche que le poste en question est équipé de MySQL et d'un serveur WAMP/MAMP.

1.3.1 Installation de la base de données

Tous les codes relatifs à la base de données sont présents dans le fichier **telephonie/database** de l'archive qui vous a été remise. Ce répertoire contient alors plusieurs types de fichiers :

- ⊙ **DB.php** Ce fichier correspond à la classe PHP développée pour réaliser la connexion à la base de données et l'exécution des requêtes/procédures vers cette base de données. Cette classe sera détaillée dans la Section 2.
- ⊙ **create_table.sql** Les requêtes SQL permettant de recréer la structure des tables SQL.
- ⊙ **.*_procedures.sql** Ces fichiers contiennent pas groupes thématiques (utilisateurs, téléphones, achats,...) les procédures à stocker dans la base de données, ainsi que des instructions pour remplir les différentes tables de la base de données.
- ⊙ **update_DB.sh** Ce script permet de lancer « automatiquement » l'installation de la base de données à partir des procédures stockées et de l'extraction de la structure de la base de données proposée dans le fichier **create_table.sql**.

Ainsi, pour installer la base de données, seule deux étapes sont nécessaires :

- ⊙ **Première étape** Ouvrir le fichier **update_DB.sh** et adapter les informations de connexion à la base de données en fonction de votre propre installation (nom de la base, mot de passe,...). La ligne à considérer est la suivante (ici, pas de mot de passe pour **root** par exemple) :

```
1  mysql --default-character-set=utf8 -h localhost -u root --password=''
    telephonie < temp.sql
```

- ⊙ **Seconde étape** Lancer ce même script (**update_DB.sh**) une fois que les informations remplies correspondent à votre configuration. Normalement, la base de données devrait se remplir automatiquement, en créant la structure des tables, en enregistrant les procédures qui doivent l'être et en se remplissant avec un ensemble de données test. Si toute fois l'installation a échoué, veuillez importer directement dans votre base de données le fichier **temp.sql** qui vient d'être généré et qui contient toutes les commandes nécessaires à l'import de la base.

1.3.2 Installation du code source

Extraction de l'archive Une fois l'installation de la base de données réalisée, il est alors nécessaire d'installer le code source de l'application sur votre poste afin de pouvoir tester l'interface développée pour ce projet.

Pour se faire, il vous faudra extraire le dossier **telephonie** de l'archive qui a été remise dans le dossier **www** correspondant à votre installation locale de WAMP/MAMP. Par exemple pour un poste équipé de Windows, ce dossier se situe généralement à **C:\wamp\www**.

Téléchargement d'un module Le code fourni est cependant incomplet, et ne permettra pas de gérer la génération du PDF de facture pour un client. En effet, nous avons pour cela utilisé une bibliothèque supplémentaire. Cependant, le code de cette dernière faisant 13Mo, nous n'avons pu l'inclure à cette archive, sans quoi nous ne pouvions plus la poster sur **pedagogie** (limite élève à 7,5Mo).

Ainsi, nous vous proposons de récupérer le contenu de cette bibliothèque directement sur le GitHub du projet :

https://github.com/stity/projet_sql/tree/master/vue

Le fichier à récupérer est le fichier **fop**, à inclure dans le dossier **vue** de l'application.

Il est à noter que ce module sert uniquement pour la génération de PDF, et n'est pas utilisé pour le reste de la plateforme...

1.3.3 Test de bon fonctionnement

Une fois le code installé et la base de données préparée, vous êtes désormais prêt pour tester l'application! Pour cela, lancer votre serveur WAMP/MAMP, et rendez vous à l'adresse suivante :

`http://localhost/telephonie/index.php`

Si vous voyez l'écran de la Figure 1 s'afficher, c'est que la procédure d'installation a correctement marché (à minima, vu qu'on ne teste pas ici toute la base de données!).



FIGURE 1 • Écran d'accueil de l'application

2 • Élaboration de la base de données

2.1 | Réalisation de l'analyse Entité-Relation

2.1.1 Entités

À la lecture de l'énoncé, nous avons extrait les entités suivantes des paragraphes suivants.

Utilisateur Chaque client doit être enregistré dans la base de données et dispose des champs suivants :

- ⊙ *Nom* Son nom.
- ⊙ *Adresse* Une adresse de facturation.
- ⊙ *Mail* Une adresse mail.
- ⊙ *Mot de passe* Pour identifier à la connexion.

Achat Les achats permettent de représenter les achats de téléphone, de formules ou des deux. Ils sont constitués :

- ⊙ *Date* Celle de l'achat.
- ⊙ *Formule* La formule achetée.
- ⊙ *Téléphone* Le téléphone acheté.
- ⊙ *Utilisateur* L'utilisateur qui a fait l'achat.

Consommation Une consommation va identifier l'ensemble des consommations effectuées par un utilisateur avec un forfait sur une période donnée.

- ⊙ *Date début* Date de début de la période d'enregistrement (de durée un mois).
- ⊙ *Consommation Données* Le montant de données consommées sur le mobile.
- ⊙ *Achat* L'achat avec lequel a été réalisé cette consommation.

SMS et MMS Les SMS et les MMS envoyés par les utilisateurs ont les propriétés suivantes :

- ⊙ *Date* La date et l'heure d'envoi.
- ⊙ *Destination* Pays de destination.
- ⊙ *Volume* Le nombre de SMS/MMS correspondant au message envoyé.
- ⊙ *Consommation* La consommation à laquelle ce message est rattaché.

Appel Les appels ont des propriétés similaires aux SMS et MMS, mais avec quelques détails supplémentaires :

- ⊙ *Date* La date et l'heure du début de l'appel.
- ⊙ *Consommation* La consommation à laquelle cet appel est rattaché.
- ⊙ *Durée* La durée de l'appel.
- ⊙ *Destination* Pays de destination de l'appel.

Facture Les factures permettent de regrouper les éléments financiers relatifs à la facturation à partir des consommations. On retrouve ainsi les propriétés suivantes :

- ⊙ *Prix* Le montant de la facture.
- ⊙ *Consommation* La consommation à laquelle cette facture est rattachée.
- ⊙ *Payé* Si le client a payé ou non cette facture.

Téléphone Cette entité permet de représenter les téléphones proposés par l'opérateur téléphonique. On retrouve alors les différentes propriétés permettant de caractériser ces produits :

- ⊙ *Écran* Type et de la taille de l'écran.
- ⊙ *RAM* La RAM du téléphone.
- ⊙ *Technologie Internet* Quelle technologie est utilisée par l'appareil.
- ⊙ *Stockage* Capacité de stockage maximale.
- ⊙ *TV* De même pour la TV.
- ⊙ *Carte SD* Possibilité de mettre une carte SD.
- ⊙ *Appareil Photo* Résolution et type d'appareil photo.
- ⊙ *Double SIM* Possibilité de faire du double

SIM.

- ◉ **Photo** L'URL présentant une photo de l'appareil.

- ◉ **Modèle** Le modèle de l'appareil.
- ◉ **Marque** La marque de l'appareil.
- ◉ **Prix** Son prix de vente.

Formule Les formules permettent de représenter les forfaits « classiques » disponibles à la vente. Ces forfaits sont caractérisés par :

- ◉ **Nom** Le nom du forfait.
- ◉ **Prix** Le prix du forfait.
- ◉ **Limite d'appel, SMS et data** Une limite pour les SMS, les appels (en durée) et les données internet (en Mo).
- ◉ **Plage horaire** Plage dans laquelle les communications sont gratuites.
- ◉ **Prix hors forfait** Différent pour les appels, les SMS et les données.
- ◉ **Bloqué** Si le forfait est bloqué ou non.

Forfait étranger En plus des forfaits usuels, il est demandé de pouvoir gérer les forfaits vers l'étranger. Pour cela, on introduit l'entité « Forfait étranger », qui sera définie par les propriétés suivantes :

- ◉ **Champs d'une formule** Tous les champs d'une formule « classique » permettent aussi de spécifier forfait étranger.
- ◉ **Destination** La zone géographique correspondant à ce forfait étranger.

Promotion Une promotion permet de fournir un rabais par rapport à une formule de base. Elle peut également être associée à un/des téléphones

- ◉ **Prix de base** Le prix de la formule initiale.
- ◉ **Nouveau prix** Le prix après rabais.
- ◉ **Date de début** Début de la promotion, éventuellement nulle si promotion « permanente ».
- ◉ **Date de fin** Fin de la promotion, éventuellement nulle si promotion « permanente ».

Plage horaire Les plages horaires permettent de découper la semaine pour adapter des forfaits à certains besoins en plage horaires.

- ◉ **Nom** Nom associé à la plage horaire pour plus de clarté.
- ◉ **Heure de début** Heure de début pour les jours concernés.
- ◉ **Heure de fin** Idem pour l'heure de fin.
- ◉ **Jour** Les jours de la semaine concernés (sous forme de plages éventuellement).

Zone géographique Pour définir les grandes zones géographiques desservies par l'opérateur.

- ◉ **Nom** Le nom de la zone.

Pays Pour définir les différents pays desservis par l'opérateur.

- ◉ **Nom** Le nom du pays.

2.1.2 Les Relations

Relations 1..N La plupart de ces relations ont été évoquées précédemment, nous les redétaillons cependant ici :

- ◉ **Consommation - Appel** Une consommation est associée à plusieurs appels.
- ◉ **Consommation - SMS** Une consommation est associée à plusieurs SMS.
- ◉ **Consommation - MMS** Une consommation est associée à plusieurs MMS.
- ◉ **Appel/SMS/MMS - Pays** Un pays peut être associé à plusieurs de ces éléments, mais chacun d'eux ne peut être à destination (ou depuis) un seul pays donné.
- ◉ **Utilisateur - Achat** Un utilisateur peut réaliser plusieurs achats, qui n'appartiennent qu'à lui.

- ◉ **Formule - Achat** Une formule peut être utilisée dans différents achat, mais un achat ne peut concerner qu'une seule formule.

Relations N..N Ces relations sont les suivantes :

- ◉ **Formule - Plage horaire** Les plages horaires sont utilisées par plusieurs formules, qui en retour peuvent avoir plusieurs plages horaires.
- ◉ **Forfait étrangers - Plage horaire** Même principe que pour les formules et les plages horaires.
- ◉ **Formule - Forfait étranger** Une formule peut être associée à plusieurs forfaits étrangers (formules monde, Europe,...). De même, un forfait étranger peut être utilisé par plusieurs formules.
- ◉ **Formule - Téléphone** Même principe que la relation entre formule et forfait étranger.
- ◉ **Zone géographique - Pays** Si le fait que plusieurs pays appartiennent à une même zone est intuitif, l'autre sens l'est moins. L'idée est ici d'avoir des « sous-zones » plus fines. Par exemple, on peut envisager une zone Maghreb et une zone Afrique, qui auront des pays en communs. De même, le réseau satellitaire peut couvrir des pays couverts par des réseaux traditionnels.

2.1.3 Spécialisation

Comme cela a été sous-entendu, les promotions sont en fait des formules spécialisées, et apportent une « sur-couche » à ces dernières. Il s'agit de la seule spécialisation que nous avons mise en place dans cette analyse.

2.1.4 Diagramme récapitulatif

En résumé des différents éléments exposés, nous avons établi le diagramme entité-relation que vous pourrez trouver à la Figure 2.

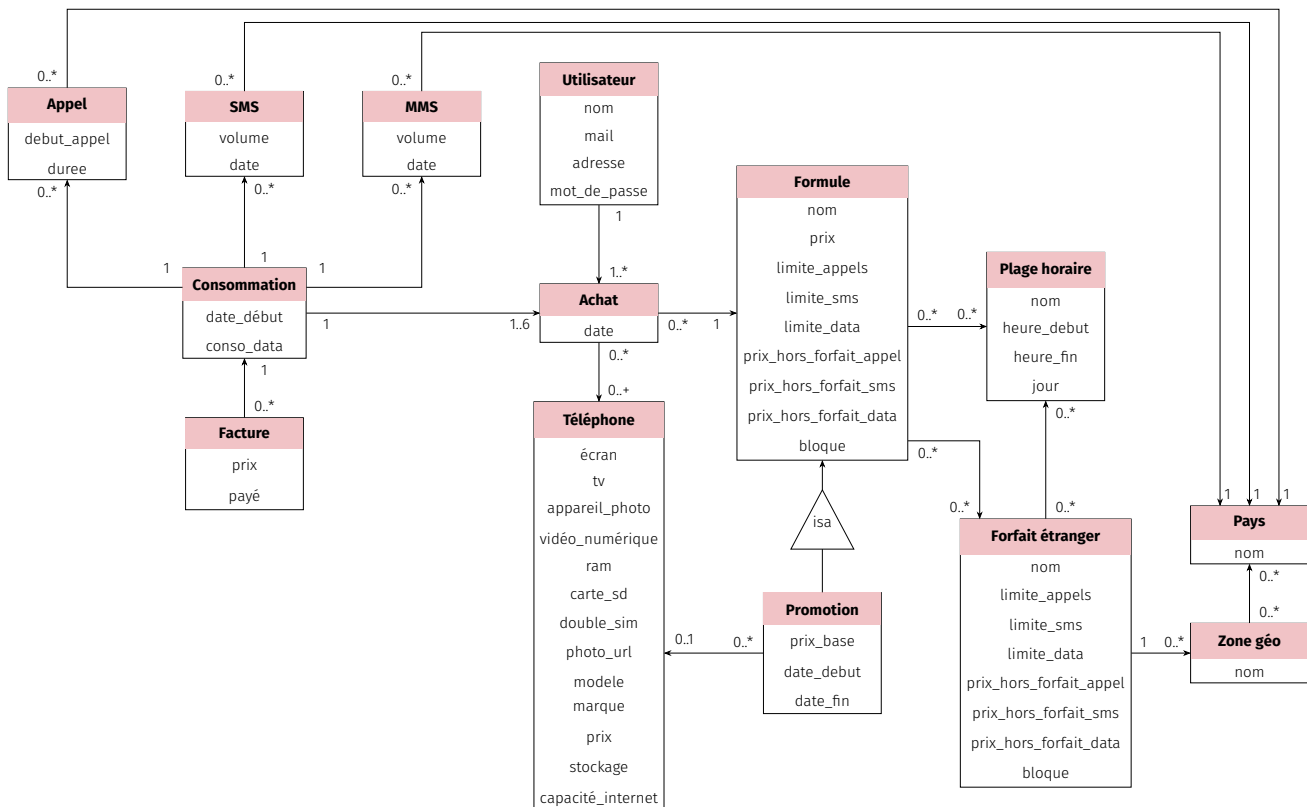


FIGURE 2 • Le diagramme Entité-Relation de notre base de données

On retrouve sur ce diagramme les différents éléments évoqués précédemment. Il nous a alors servi à définir les tables que nous désirions utiliser pour notre base de données.

2.2 Définition des tables à utiliser

2.2.1 Gestion des relations $N . . N$

Pour ces relations, nous créons systématiquement une table par entité et une table de couples de clés provenant de ces tables pour réaliser la jonction entre les deux notions.

Ces tables seront mentionnées sous la forme `nom_entité_1_nom_entité_2`.

2.2.2 Gestion des relations $1 . . N$

Pour ces relations, on crée une table par entité. De plus, pour la table provenant de l'entité ne pouvant être associé qu'à un élément de l'autre entité, on ajoute une colonne contenant la clé d'une valeur de l'autre table.

Par exemple pour entre `achat` et `formule`, on trouvera dans la table `achat` une colonne `id_formule`.

2.2.3 Gestion de la généralisation *isa*



Dans le cadre de ce projet, nous avons choisi de transformer cette généralisation via une table regroupant les entités « Formule » et « Promotion », mais avec des champs avec des valeurs par défaut de -1 ou NULL pour différencier les éléments.

De plus, ceci nous a permis de créer des formules associées à des téléphones, ce qui a enrichi le modèle.

2.2.4 Diagramme final

Ainsi, nous avons en définitive aboutit au diagramme de la Figure 3, qui présente toutes les tables proposées, ainsi que leurs liens.

La légende de cette figure est la suivante :

- ⊙  Il s'agit des clés primaires des tables. Si une table dispose de plus d'une clé primaire, alors c'est la valeur de la paire, du triplet,...qui fait office de clé primaire.
- ⊙  Il s'agit des champs utilisant la valeur d'une clé primaire d'une autre table (clé étrangère).

De plus, les tables ayant des liens entre elles (ie. utilisation de clés étrangères), sont reliés dans le schéma de la Figure 3.

On remarquera en particulier sur cette figure, le fait que l'entité promotion ait été « comprise » dans `formule`, avec les champs `formule_base`, `date_debut` et `date_fin` qui soient pris en compte dans cette table.

Une dernière remarque concerne les champs `is_deleted` des tables `formule` et `forfait_etrananger`. Ils ont été mis en place pour persister dans la base de données les formules n'étant plus disponibles à la vente, mais qui restent utilisées par des utilisateurs.

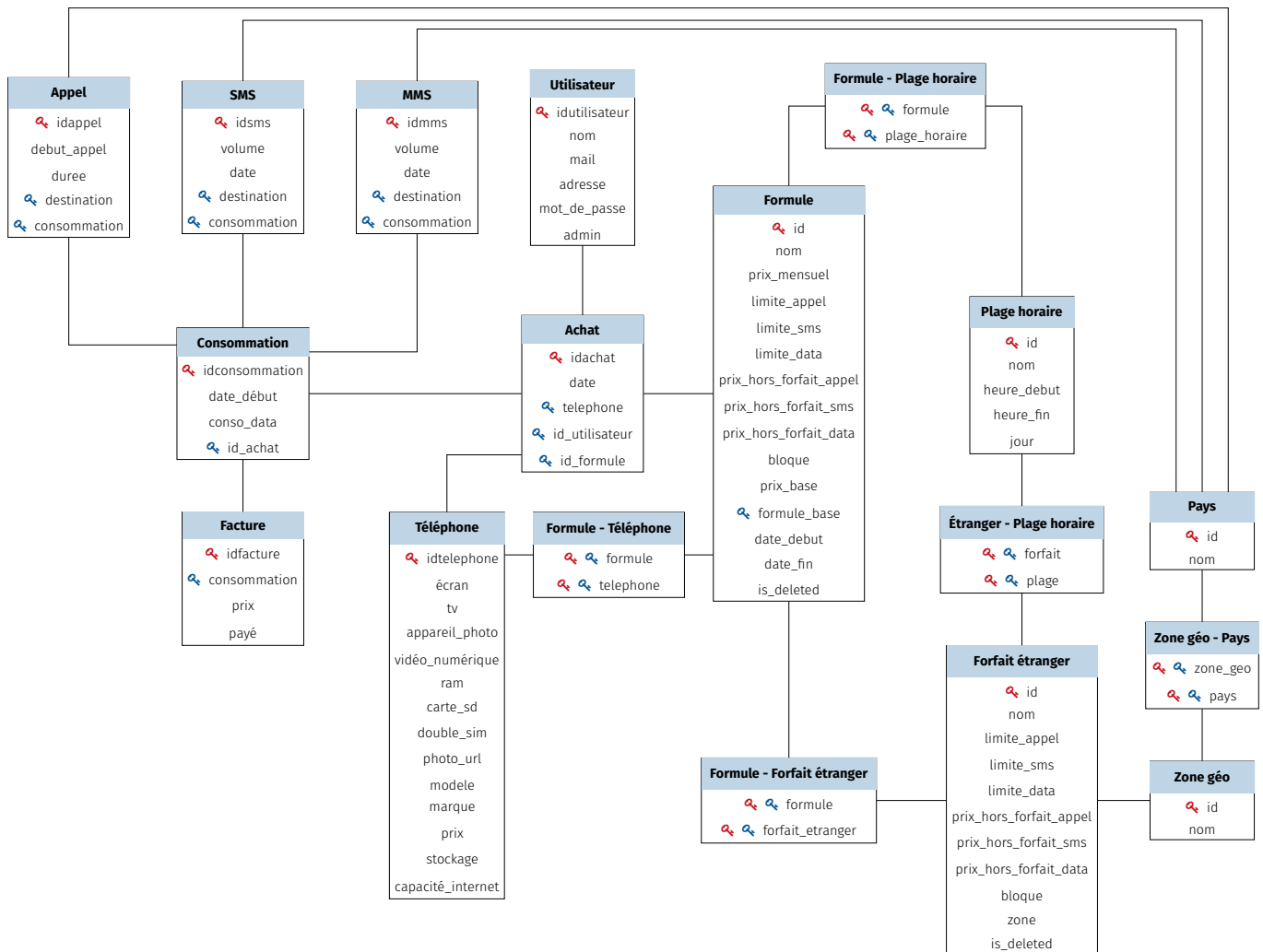


FIGURE 3 • Relations entre les tables de la base de données

3 • Plateforme réalisée

Nous détaillerons dans cette partie les différents visuels de la plateforme. Nous préciserons également le fonctionnement administratif de la plateforme (clients, administrateurs, visiteurs).

3.1 | Les différentes vues

3.1.1 Liste des vues

La plateforme est organisée selon une architecture MVC (Modèle - Vue - Contrôleur). Ainsi, les différents visuels/pages proposés (les vues) sont chacun représentés par un fichier. Dans notre cas, ces fichiers sont situés dans le répertoire `telephonie/vue`. Nous allons dans un premier temps détailler la liste des vues, ces dernières seront ensuite présentées (fonctionnement et interaction avec les utilisateurs) dans les sous-sections suivantes.

Ainsi, l'application propose au total sept vues :

- ◉ **Accueil** La page d'arrivée dans l'application qui présente uniquement les principaux objectifs de ce dernier. Ce visuel vous a été présenté à la Figure 1.
- ◉ **Téléphones** Cette page présente les différents téléphones proposés à la vente par Centrale-Télécom. Elle permet au client de s'informer sur les modèles et de les acheter. Dans le même temps, elle permet à l'équipe administrative d'en référencer de nouveaux.
- ◉ **Compte/Utilisateur** Ces vues permettent soit l'administration des utilisateurs par les administrateurs, soit à un client donné de consulter ses informations et de changer ses informations.
- ◉ **Connexion** Une vue permet aux utilisateurs de se connecter/déconnecter.
- ◉ **Factures** Cette vue n'est accessible qu'aux clients et permet à ces derniers de consulter leurs factures et de télécharger cette dernière au format PDF.
- ◉ **Abonnements** Cette vue se décompose en trois vues intermédiaires :
 - ▷ **Mes abonnements** Cette vue permet à un client de retrouver tous les abonnements auxquels il a souscrit, ainsi que les téléphones achetés chez Centrale-Télécom.
 - ▷ **Vers l'étranger** Cette vue permet de préciser tous les abonnements étrangers proposés par Centrale-Télécom. Elle permet également aux administrateurs d'ajouter de nouveaux forfaits.
 - ▷ **Formules** Cette vue permet de préciser toutes les formules proposées par Centrale-Télécom. Elle permet également aux administrateurs d'ajouter de nouvelles formules.

Les spécificités par utilisateurs seront précisées plus en détails dans les sous-sections suivantes, ainsi que toutes les actions possibles.

3.1.2 Navigation dans l'application

Le dernier point général sur cette application est la navigation. Cette dernière est facilitée par un panneau latéral qui présente tous les menus de l'application. Dans le cas d'un client « lambda », on dispose du panneau latéral observable à la Figure 4.

À moins d'être à l'accueil (cas de la Figure 4), la page courante est surlignée dans le volet de navigation pour que les utilisateurs sachent où ils se situent dans l'application.

3.2 | Gestion des utilisateurs

3.2.1 Les différents types d'utilisateurs

L'application est structurée autour de trois niveaux hiérarchiques pour les utilisateurs :

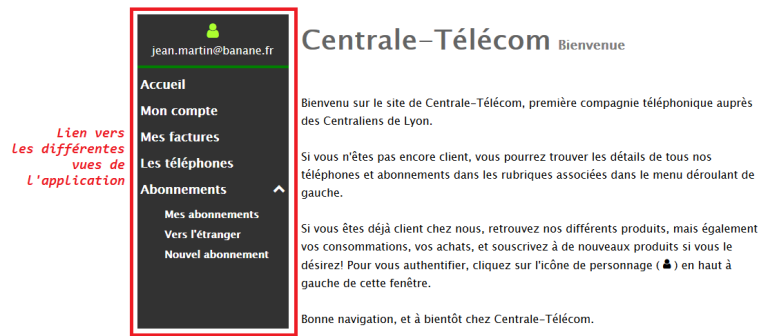


FIGURE 4 • Panneau de navigation pour l'application

- ◉ **Visiteur** Il s'agit du niveau avec le moins d'accréditations possibles. Cet utilisateur correspond à une personne lambda naviguant sur le site, sans être client de Centrale-Télécom.
- ◉ **Client** Ce niveau permet de représenter les clients qui viennent consulter leurs factures, leurs abonnements ou encore les nouvelles offres disponibles.
- ◉ **Administrateur** Ce grade est celui avec le plus de droit. Il représente les employés de Centrale-Télécom qui ont donc le droit d'éditer les profils, créer des abonnements, des téléphones,...


Les clients et les administrateurs sont des profils qui sont détectés après une connexion à la plateforme. Ils sont différenciés au niveau de la base de données via la colonne **admin** de la table **utilisateur**, qui est mise à **TRUE** si l'utilisateur est un administrateur. Une fois la connexion effectuée, on stocke les informations dans la variable globale **\$_SESSION**. Le cas des visiteurs correspond alors aux valeurs par défaut pour **\$_SESSION**. On retrouve alors les caractéristiques de la Table 1.

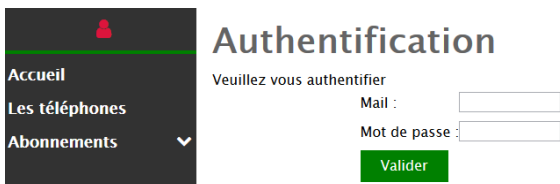
Type d'utilisateur	Champ log_in	Champ login_level
Visiteur	False	Sans importance (False)
Client	True	False
Administrateur	True	True

TABLE 1 • Valeurs stockées dans **\$_SESSION** pour les différents profils d'utilisateurs

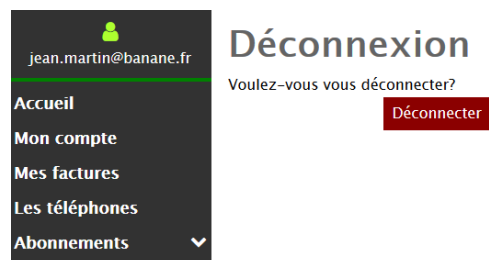
Ces grandeurs nous permettent d'adapter les affichages en fonction du type d'utilisateur.

3.2.2 Connexion et déconnexion

Réalisation des actions Ces deux actions sont réalisées à partir d'une vue accessible en cliquant sur l'icône .



(a) Connexion d'un utilisateur






(b) Déconnexion d'un utilisateur

FIGURE 5 • Les fenêtres de connexion et déconnexion des utilisateurs

Une première vue (Figure 5a) permet aux utilisateurs de se déconnecter. Le principe est usuel, où l'utilisateur va fournir son adresse mail (qui fait office d'identifiant ici) ainsi que son mot de passe. Si les deux concordent, l'utilisateur est connecté, sinon il devra ré-essayer.

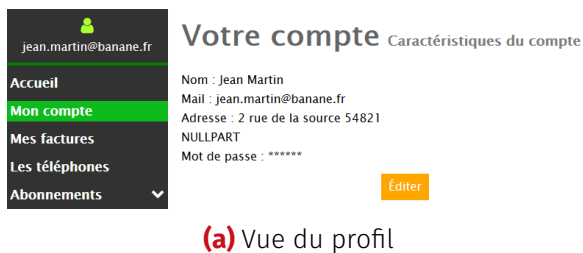
Pour la déconnexion, il suffit de se rendre aussi sur cette page (Figure 5b), qui propose alors un bouton pour se déconnecter.

Différences visuelles On remarquera également une différence sur la partie haute de la barre de navigation, où l'avatar  est en rouge () si l'utilisateur n'est pas enregistré et en vert () si l'utilisateur est connecté. De même, quand l'utilisateur est connecté, son adresse mail apparaît sous cette icône.

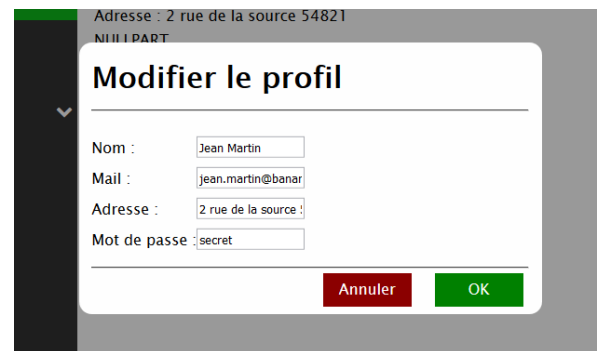
Pour les visiteurs Un visiteur ne disposant pas de compte, il ne peut se connecter, et n'aura donc accès à aucune vue de celles précisées dans la suite de cette section.

3.2.3 Édition du profil en tant que client

Un client dispose de la possibilité de voir et éditer les informations de son profil (sauf son statut!) directement par la vue intitulée « Mon compte ». La vue de consultation du profil vous est proposée à la Figure 6a présente les différentes informations au client, ainsi qu'un bouton d'édition. La vue d'édition associée est présentée à la Figure 6b.



(a) Vue du profil




(b) Édition de son profil

FIGURE 6 • Gestion du compte par un client

3.2.4 Édition des profils en tant qu'administrateur

Un administrateur dispose lui de plus de droit. Il peut ainsi voir la liste de tous les utilisateurs inscrits à la plateforme, qu'ils soient administrateurs ou simples clients. On remarquera cependant qu'un administrateur qui consulte cette liste ne peut se voir lui-même, pour éviter une modification involontaire de son profil. Cette vue vous est proposée à la Figure 7a

En effet, outre la liste des utilisateurs, l'administrateur dispose du droit de modifier tous les comptes, que ce soit pour les adresses mails, les noms, les mots de passe ou encore les droits d'accès à la plateforme (administrateur ou non). La fenêtre d'édition est précisée à la Figure 7b et s'ouvre en cliquant sur une icône .

Enfin, l'administrateur peut également supprimer des comptes clients en cliquant sur l'icône . On remarquera que dans cette vue, des informations sont apportées à l'administrateur en cas d'échec ou de réussite de sa manipulation.

3.3 Les téléphones

Cette partie permet de voir la liste des téléphones, éditer leurs configurations et les acheter.

Les utilisateurs Liste des utilisateurs

En tant qu'administrateur de la plateforme, vous pouvez supprimer les différents comptes des clients ou éditer leurs propriétés.

Nom	Mail	Adresse	Mot de passe	Administrateur
Jean Martin	jean.martin@banane.fr	2 rue de la source 54821 NULLPART	secret	<input checked="" type="checkbox"/> <input type="checkbox"/>
Jean Ramos	jean.ramos@hotmail.fr	4 impasse des muguets 15541 Champoux	ramos69	<input checked="" type="checkbox"/> <input type="checkbox"/>
Roger De La Chance	rogerdelachance@free.fr	4bis rue du trefle 75421 Ferrache	chanceux77713	<input checked="" type="checkbox"/> <input type="checkbox"/>

(a) Liste des clients

Modifier le profil

Nom :
 Mail :
 Adresse :
 Mot de passe :
 Administrateur : ☐ Oui ☒ Non

(b) Édition d'un utilisateur

FIGURE 7 • Gestion des comptes clients par un administrateur

3.3.1 Consulter la liste des téléphones

Cette action est disponible pour tous les types d'utilisateurs. Pour cela, il suffit de cliquer sur la lignes « Les téléphones » de la barre de menu. On aboutit alors à l'interface de la Figure 8.

Les téléphones Liste des téléphones

Cliquez sur les images pour plus de détails sur les téléphones.

Marque	Modèle	Prix	RAM	Visuel
Crosscall	Trekker M1 Core	300€	2Go	 Acheter
LG	G4	275€	3Go	 Acheter

FIGURE 8 • Consulter la liste des téléphones


On remarquera que cette liste ne propose pas initialement toutes les caractéristiques des téléphones. Afin d'accéder à ces dernières, l'utilisateur doit cliquer sur l'image du téléphone. Une modale avec tous les détails apparaît alors, comme proposée à la Figure ??.


3.3.2 Créer et modifier un téléphone

Ces actions ne peuvent être réalisées que par un administrateur.

Créer un nouveau téléphone Pour cela, un administrateur doit simplement cliquer sur le bouton « Nouvel appareil ». Une modale s'ouvre alors, lui proposant de remplir tous les champs permettant de caractériser le téléphone. Cette modale vous est proposée à la Figure 10.

Si en cliquant sur « OK », la table n'est pas mis à jour, c'est que certains champs obligatoires n'ont pas été remplis. Sinon, tous les champs remplis sont corrects.

Édition d'un téléphone De manière transparente, l'administrateur peut cliquer sur l'icône  d'un téléphone. Une modale de même aspect que celle de la Figure 10 apparaît alors avec les valeurs pré-remplies du téléphone. L'administrateur peut alors éditer ces valeurs à loisir.

Suppression d'un téléphone Cette dernière se fait en cliquant sur l'icône  du téléphone. Ce dernier n'est alors pas supprimé entièrement de la base de données. Il est supprimé des formules qui le proposait

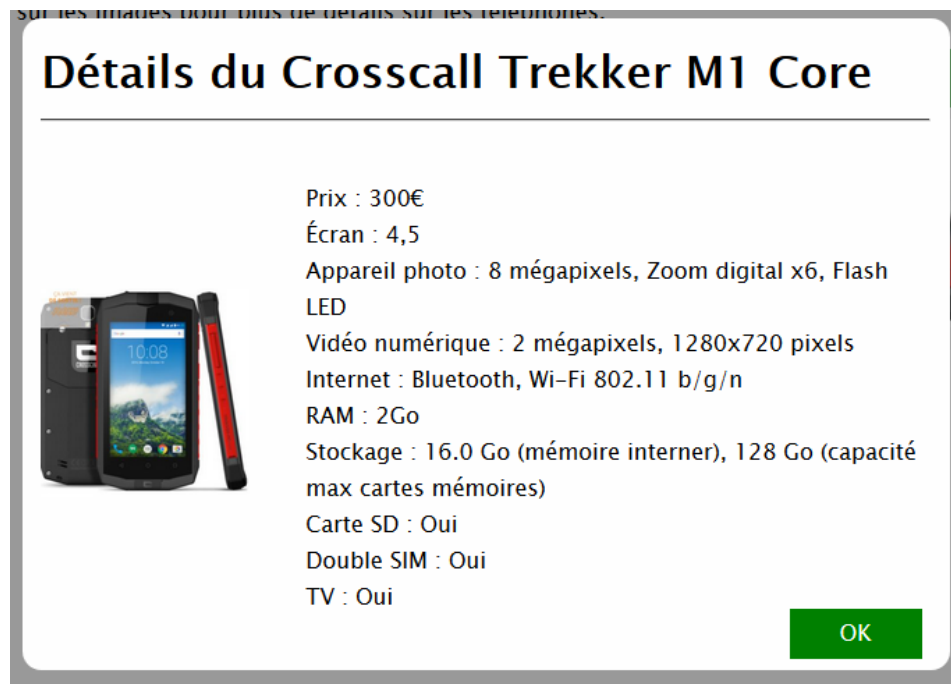


FIGURE 9 • Affichage des détails d'un téléphone

auparavant, et son champ `is_deleted` qui représente sa mise sur le marché est mis à **TRUE**, pour que ce dernier n'apparaisse plus dans les listes de téléphones. Il n'est cependant pas totalement supprimé de la base de données, pour que les utilisateurs l'ayant acheté le voit toujours apparaître dans leur liste d'achat.

3.3.3 Acheter un téléphone

Cette action ne peut être réalisée que par un client. Pour cela, ce dernier dispose de la possibilité de cliquer sur la case « Acheter » à la fin de chaque ligne de la table présentant les téléphones. En faisant ainsi, une modale s'ouvre pour valider la commande, comme présentée à la Figure 11.

En cliquant sur « OK », le client valide alors la commande, il peut annuler l'action en cliquant sur « Annuler ». La consultation de la liste des téléphones achetés sera détaillée plus loin dans la partie consacrée aux abonnements.

3.4 Consultation des factures

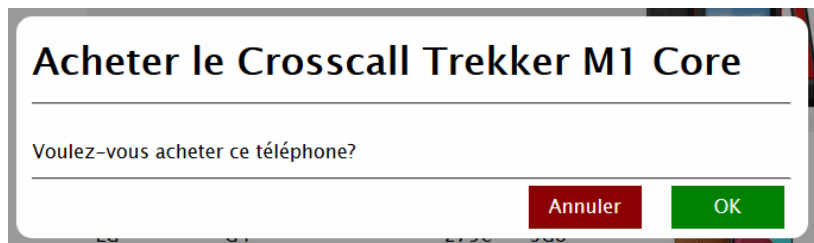
Cette section concerne la création et la consultation de factures ainsi que la génération des détails de consommation de chaque utilisateur sous forme d'un fichier PDF.

3.4.1 Liste des factures

Lors que l'utilisateur identifié clique sur la section « Les factures » il voit alors apparaître la liste de ses factures dans un tableau ressemblant à celui présent sur la Figure 12

Les champs de ce tableau sont les suivants :

- ◉ **Date** Le mois de facturation : correspond à la période pendant laquelle les opérations téléphoniques ont eu lieu.
- ◉ **Montant** Le montant de la facture à payer en euros.
- ◉ **Reglé** « Oui » si la facture est déjà payée, « Non » sinon.
- ◉ **Détails** Un lien vers une page qui génère et affiche le détail des consommations de l'utilisateur pour le mois sélectionné.

**FIGURE 10** • Création d'un téléphone**FIGURE 11** • Confirmation de l'achat d'un téléphone

3.4.2 Détail des consommations

Le PDF généré donne les consommations détaillées de l'utilisateur pour trois types de consommations différentes :

- ◉ *SMS* Présenté en Figure 13a
- ◉ *MMS* Présenté en Figure 13b
- ◉ *appel* Présenté en Figure 13c

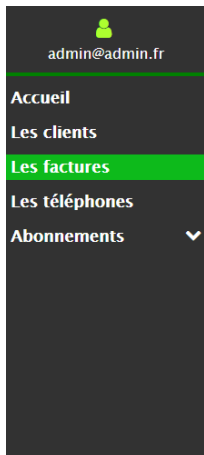
3.5 | Gestion des abonnements

Cette partie va présenter toutes les fonctionnalités relatives à la consultation, la modification et la souscription à un abonnement. Du point de vue des utilisateurs, tous (y compris les visiteurs) ont accès à la consultation des offres (formules et forfaits étrangers). Par contre, un client a accès en plus à un résumé de ses achats (téléphone et formules).

3.5.1 La liste des formules

Cette vue est la principale associée à la gestion des abonnements. Elle présente un résumé de la liste des formules proposés par Centrale-Télécom. Le tableau présente donc les principales caractéristiques des formules, à savoir :

- ◉ *Nom et tarif* Les informations basiques pour que les utilisateurs se repèrent dans les formules.
- ◉ *Promotion* Cette valeur est mise à « Oui » si la formule est une promotion, « Non » sinon.



Factures Consulter ma facture

Votre facture est directement consultable au format PDF!

Date	Montant	Réglé	Détails
Oct 2016	62.5€	Non	
Nov 2016	72.59€	Non	
Dec 2016	35.59€	Non	

FIGURE 12 • Liste des factures

Facture de téléphone

Liste des sms

Date	Volume	Destination	Zone
02/11 à 12h42	6	Rwanda	Reste du monde
03/11 à 12h43	2	Grande Bretagne	Europe/DOM/COM
03/11 à 19h21	8	Portugal	Europe/DOM/COM
03/11 à 19h48	5	Tunisie	Maghreb
04/11 à 16h35	5	France	France
05/11 à 18h28	2	Thaïlande	Asie
06/11 à 20h28	4	Italie	Europe/DOM/COM
07/11 à 19h04	2	France	France
08/11 à 16h05	8	Brazil	Amérique du Sud
08/11 à 18h43	1	Russie	Europe élargie

(a) Liste des SMS

Facture de téléphone

Liste des MMS

Date	Volume	Destination	Zone
01/11 à 18h38	3	Argentine	Amérique du Sud
03/11 à 18h57	7	Argentine	Amérique du Sud
04/11 à 10h08	2	Arabie Saoudite	Reste du monde
04/11 à 10h41	9	Corée du Sud	Asie
04/11 à 15h30	2	Mexique	Amérique du Nord
04/11 à 19h52	8	Pôle Nord	Réseaux satellites
05/11 à 11h46	3	Egypte	Maghreb
08/11 à 10h22	8	Italie	Europe/DOM/COM
08/11 à 20h04	6	Norvège	Europe élargie
09/11 à 17h54	1	Ukraine	Europe élargie
10/11 à 12h00	5	Pologne	Europe élargie
11/11 à 13h50	9	Egypte	Maghreb
12/11 à 11h49	8	Arabie Saoudite	Reste du monde
12/11 à 12h00	7	Indonésie	Asie
12/11 à 18h25	8	Estonie	Europe élargie
12/11 à 20h30	3	Espagne	Europe/DOM/COM
16/11 à 10h49	3	Norvège	Europe élargie

(b) Liste des MMS

Facture de téléphone

Liste des appels

Date	Durée	Destination	Zone
01/11 à 12h49	00:28:57	Arabie Saoudite	Reste du monde
03/11 à 12h23	00:20:26	Bolivia	Amérique du Sud
04/11 à 14h41	00:07:45	Norvège	Europe élargie
06/11 à 15h10	00:20:39	Thaïlande	Asie
08/11 à 10h30	00:01:17	Maroc	Maghreb
08/11 à 12h17	00:05:34	Pologne	Europe élargie
09/11 à 10h29	00:25:24	Cuba	Amérique du Nord
09/11 à 12h46	00:02:19	Estonie	Europe élargie
09/11 à 14h49	00:22:55	Terre Adèle	Réseaux satellites
10/11 à 10h01	00:24:12	Pérou	Amérique du Sud
10/11 à 13h30	00:19:05	Ukraine	Europe élargie
11/11 à 11h23	00:28:51	Mexique	Amérique du Nord
11/11 à 11h54	00:06:13	Syrie	Reste du monde
11/11 à 15h21	00:29:06	Rwanda	Reste du monde
11/11 à 15h37	00:12:44	Ukraine	Europe élargie
12/11 à 11h57	00:19:53	Chine	Asie
12/11 à 13h09	00:17:16	Norvège	Europe élargie

(c) Liste des appels

FIGURE 13 • Listes générées dans le PDF

- ◉ **Téléphone associé** L'icône permet de savoir si la formule est associée à un ou plusieurs téléphones. Ceci permet de représenter les liens (par exemples promotionnels) entre une formule et un téléphone. Si un ou deux téléphones sont associés, alors l'icône est , sinon il s'agit de .
- ◉ **Forfaits étrangers** Tout comme le champs sur les téléphones, il permet de savoir si l'offre est associée à une formule vers l'étranger. Ici aussi, si un ou plusieurs forfaits étrangers sont inclus dans la formule, alors l'icône est , sinon il s'agit de .
- ◉ **Boutons d'édition** Ces boutons ne sont accessibles qu'en tant qu'administrateur.

Ces différents éléments peuvent être observés à la Figure 14 qui propose la situation en tant qu'administrateur (avec les boutons d'édition et de création de formule).

admin@admin.fr

Accueil

Les clients

Les factures

Les téléphones

Abonnements

Vers l'étranger

Les abonnements

Les abonnements Édition des abonnements

Cliquez sur le nom de l'abonnement pour plus de détails.

Vos filtres :

☒ Abonnements
 Prix minimum : €/mois

☒ Promotions
 Prix maximum : €/mois

Chercher

Nom	Tarif (€/mois)	Promotion	Téléphone associé	Forfaits étrangers
Formule Larnaque	20	Non		
Formule Deluxe	20	Non		
Formule Debaze	50	Non		
Super Promo	10	Oui		

Nouvel abonnement

FIGURE 14 • Vue présentant la liste des formules de Centrale-Télécom

Dans la suite, nous allons détailler les différentes actions possibles dans cette vue.

3.5.2 Consultation des détails d'une formule

Détails généraux Si la table présentée à la Figure 14 permet de visualiser les fondamentaux des formules, plus de détails sont disponibles pour ces dernières. Ainsi, en cliquant sur le nom de la formule (par exemple sur « Super Promo »), on obtient une fenêtre modale qui présente les détails de la formule, comme proposé à la Figure 15.

FIGURE 15 • Détails d'une formule

On remarquera en particulier que cette formule est une promotion sur la « Formule Larnaque » et que les informations initiales de cette formule sont remontées.

Cette fonctionnalité est accessible à tous les utilisateurs.

Téléphones associés à la formule En cliquant sur l'icône 📱, on peut consulter la liste des téléphones qui sont associés à la formule. Ces téléphones peuvent par exemple bénéficier de tarifs préférentiels si on achète cette formule (ceci n'a pas été matérialisé dans la base de données cependant). Un exemple vous est proposé à la Figure 16a, où on peut visualiser que la « Formule Larnaque » est associée à un téléphone.

On remarquera que cette vue est uniquement consultative, et que pour plus de détails, il est nécessaire de se rendre à la page de description des téléphones.

Forfaits étrangers associés à la formule Pour obtenir la liste des forfaits étrangers associés à une formule, il suffit (de manière similaire aux téléphones associés), de cliquer sur l'icône 🌐 qui va alors ouvrir une modale avec la liste des forfaits étrangers compris dans la formule. Un exemple vous est proposé à la Figure 16b.

(a) Téléphone(s) associé(s)

(b) Forfait(s) étranger(s) associé(s)

FIGURE 16 • Consultation d'éléments associés à une formule

Recherche avancée Afin de pouvoir plus facilement cibler une formule, nous avons mis en place quelques éléments de recherche pour les principaux critères. Il est ainsi possible de lancer une recherche en spécifiant un prix minimum et un prix maximum. Il est également possible de filtrer les résultats selon qu'il s'agisse d'offres promotionnelles, ou d'offres non-promotionnelles ou des deux.

Sur la base de tests qui vous a été fournis, on peut par exemple demander les offres non promotionnelles ayant un tarif entre 15 et 25 euros. On obtient alors le résultat de la Figure 17.

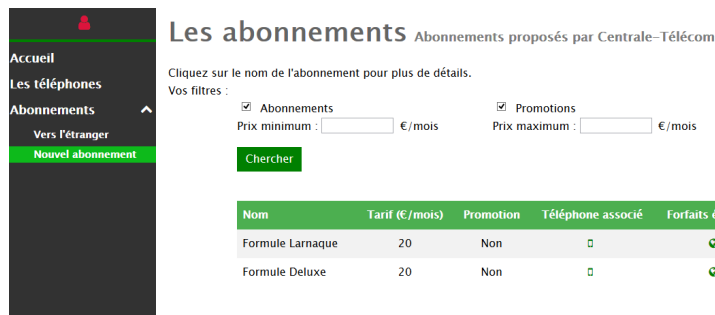


FIGURE 17 • Filtrage des formules avec l'outil de recherche

3.5.3 Consultation des forfaits étrangers

La consultation des forfaits étrangers se fait selon les mêmes modalités que les formules traditionnelles. On retrouve cependant moins de caractéristiques pour ces derniers. En effet, ils ne sont pas associés à des téléphones, ainsi, leurs principales caractéristiques sont juste des tarifs et des zones géographiques visées. La consultation de la liste est alors immédiate (Figure 18b), et les détails sur une formule sont également obtenus en cliquant sur son nom (Figure ??).

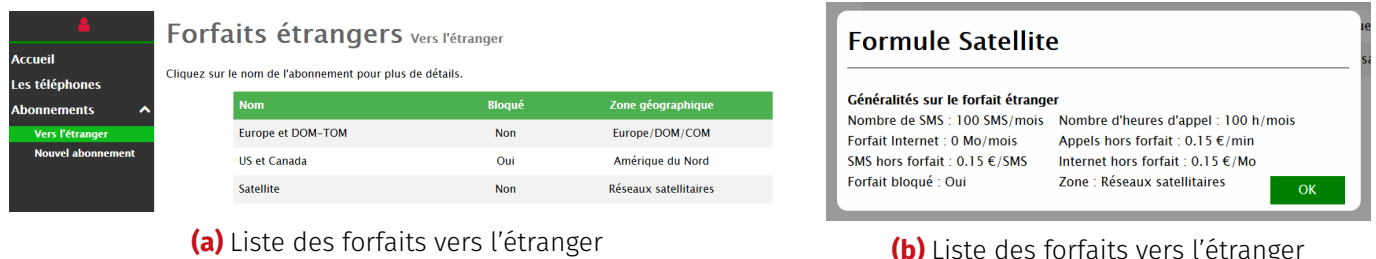


FIGURE 18 • Consultation de la liste des forfaits étrangers et de leurs détails

Ici aussi, cette consultation est accessible à tous les niveaux d'utilisateur.

3.5.4 Création, édition et suppression d'une formule ou d'un forfait étranger

Ces fonctionnalités ne sont accessibles qu'aux administrateurs. Pour cela, ces derniers doivent se rendre dans les vues listant les offres (formules ou forfaits étrangers). Pour l'exemple, nous prendrons le cas des formules (qui est un peu plus sophistiqué).

Création d'une formule Pour cela, l'administrateur doit cliquer sur le bouton « Nouvel abonnement ». La modale de la Figure 19 apparaît alors et lui permet de configurer cet abonnement.

Outre les champs classiques comme les prix ou les limites de consommation, on retrouve certains champs dépendant directement des informations stockées en base de données, comme la formule de base utilisée à partir de laquelle on applique une réduction. On remarquera l'absence des forfaits étrangers et des portables associés, qui pour des raisons de facilité de code côté utilisateur ont été déportées dans des modales particulières.

Associer des forfaits étrangers à la formule Pour cela, l'administrateur doit cliquer sur l'icône . À la différence de la modale précédemment sélectionnée, celle de l'administrateur permet de (dé)sélectionner les abonnements à inclure ou non dans la formule (Figure 20b). Le changement est immédiat s'il clique sur « OK ».

Associer des portables à la formule Le fonctionnement est le même que pour les forfaits étrangers, mais en cliquant désormais sur (Figure 20a).

Édition d'une formule Pour éditer les champs remplis à la création du forfait (voir Figure 19), l'administrateur peut cliquer sur un icône . Une modale avec les mêmes champs (mais configurés en accord avec



Créer un abonnement

Nom :

Prix mensuel :

Nb heures d'appel :

Nb SMS :

Nb Mo Internet :

Appels hors forfait (€/min) :

SMS hors forfait (€/SMS) :

Internet hors forfait (€/Mo) :

Prix avant réduction :

Forfait bloqué : ☐ Oui ☒ Non

Formule de base :

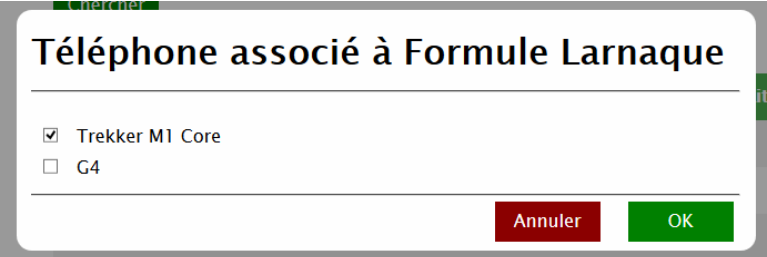
Plages horaires :

Période semaine :

Date de début :

Date de fin :

FIGURE 19 • Modale de création d'un nouvel abonnement




Téléphone associé à Formule Larnaque

☒ Trekker M1 Core

☐ G4

(a) Téléphones



Forfaits étrangers associés à Formule Larnaque

☒ Europe et DOM-TOM

☐ US et Canada

☐ Satellite

(b) Forfaits étrangers

FIGURE 20 • Modale pour associer des éléments à une formule

les caractéristiques de la formule) s'ouvre et lui permet de modifier s'il le souhaite les différentes valeurs.

Suppression d'une formule Pour cela, l'administrateur doit cliquer sur l'icône ✕ associée à la formule. À noter que la suppression n'est pas réelle, au sens où la formule n'est pas supprimée de la base de données. En effet, des clients pouvant avoir des consommations en lien avec cette formule, nous mettons en fait à jour un champs booléen `is_deleted` qui permet de savoir si une formule est supprimée, tout en conservant ses caractéristiques pour l'établissement des factures des clients. Le processus est le même pour les forfaits étrangers.

3.5.5 Souscription à un forfait et consultation des forfaits

Ces fonctionnalités ne sont actives que pour les clients, nous allons les détailler dans cette sous-section.

Souscrire à un forfait La souscription se fait de manière similaire à celle de l'achat d'un téléphone. Ainsi, pour souscrire à un abonnement, un client doit se rendre dans la liste qui détaille tous les abonnements. Il dispose alors d'une colonne avec deux valeurs possibles :

- ⦿ **Souscrire** En cliquant sur ce mot, le client souscrit alors automatiquement à la formule.
- ⦿ **Désinscrire** En cliquant sur ce mot, le client se désinscrit automatiquement de la formule.

Si ce processus est (bien trop) simplifié par rapport à la réalité, il nous permet de représenter de manière fonctionnelle le mécanisme de souscription-désinscription et d'effectuer les actions associées en base de données. Les souscriptions étant considérées comme des achats, les modifications sont réalisées sur la table **achat** (tout comme les téléphones).

Une fonctionnalité non prise en compte ici est de pouvoir associer un téléphone à cette souscription, ce qui n'a pas été prévu dans l'application finale, bien que cela soit possible en base de données (voir la procédure **addAchat**).

Consultation des forfaits Cette consultation peut s'effectuer de deux manières. Tout d'abord, dans la liste des forfaits, ceux où il est marqué « Désinscrire » sont en fait les forfaits auxquels le client a déjà souscrit. Cependant, pour plus de lisibilité, une interface spécifique est proposée au client, nommée « Mes abonnements ».

Cette interface présente alors deux informations : la liste des formules auxquelles le client a souscrit et la liste des téléphones qu'il a achetés. Comme pour les vues avec la liste des formules et des téléphones, le client peut également consulter les détails de ces achats de manières similaires (clic sur l'image du téléphone, le nom de la formule, les icône 📶 et 📶). L'interface générale vous est proposée à la Figure 21.

jean.martin@banane.fr

Accueil
Mon compte
Mes factures
Les téléphones
Abonnements

Mes abonnements
Vers l'étranger
Nouvel abonnement

Mes abonnements

Détail de vos abonnements

Vous pourrez trouver dans la table ci-dessous la liste de vos forfaits.

Nom	Tarif (€/mois)	Promotion	Téléphone associé	Forfaits étrangers
Formule Larnaque	20	Non	📶	📶

Et ici la liste de tous les téléphones achetés auprès de nos services :

Marque	Modèle	Prix	RAM	Visuel
Crosscall	Trekker M1 Core	300€	2Go	

FIGURE 21 • Vue résumant tous les achats (formules et téléphones) d'un client

Ceci permet donc au client d'avoir une vision synthétique de ses achats.

4 • Aspects techniques de la plateforme

4.1 Architecture de la plateforme

4.1.1 Architecture MVC

La plateforme a été réalisée en ayant pour volonté de mettre en place une architecture MVC minimale. Pour cela, le code a été organisé selon trois dossiers principaux :

- ⊙ **contrôleur** Ce fichier contient tous les contrôleurs associés au modèle.
- ⊙ **vue** Ce fichier contient toutes les vues du site, ie. des visuels qui réutilisent des informations calculées par les contrôleurs.
- ⊙ **database** Normalement, le fichier aurait dû être appelé **modèle** dans une architecture MVC classique, cependant, toutes les opérations liées au modèle sont presque exclusivement réalisées en SQL dans cette application, ce fichier a donc « pris sa place ».

Dans la suite, nous allons préciser comment les ressources sont chargées pour chaque page.

4.1.2 Appel des vues

Charger les fichiers Le centre de l'architecture mise en place est le fichier `index.php`. Ce dernier réalise toutes les inclusions nécessaires en PHP, via les commandes :

```
1 require 'database/DB.php';
2 require 'contrôleur/users_controller.php';
3 require 'contrôleur/bills_controller.php';
4 require 'vue/vue.php';
5 .../...
```

Création d'une vue Ensuite, nous créons dans ce fichier une vue (instance de l'objet `Vue`). Cet instance permettra d'afficher les éléments dans la page.

Assignment de la vue à un contrôleur Enfin, le fichier va récupérer via les paramètres passés dans l'URL, le nom de la vue qui est demandée. En fonction du résultat, il va créer le contrôleur associé à la vue. Ceci est réalisée par le `switch` suivant :

```
1 if (isset($_GET['vue'])){
2     switch($_GET['vue']){
3         case 'users':
4             $contrôleur = new UsersController($vue, $_SESSION['log_in'], $_SESSION['login_level']);
5             break;
6         case 'bills':
7             $contrôleur = new BillsController($vue, $_SESSION['log_in'], $_SESSION['login_level']);
8             break;
9         case .../...
```

Ainsi, chaque contrôleur est créé avec la vue pour l'afficher en paramètre. Chaque contrôleur va alors appeler spécifiquement son affichage via des commandes de la forme :

```
1 function __construct($vue) {
2     $vue->display('bills', 'Factures', 'Consulter ma facture', $this);
3 }
```

Récupération du fichier spécifique à la vue Enfin, la dernière étape revient à charger le fichier de la vue associée au contrôleur (et non plus la classe `Vue`). Pour cela, l'instance `$vue` dispose d'une méthode

`determine_vue` qui est appelée par `display` et qui contient les associations entre les vues demandées et leur fichier associé.

Relancer le processus Ce processus sera relancé à partir de la recherche du contrôleur dès que le client clique sur un des liens du panneau latéral.

4.1.3 Les fichiers affichés

Comme nous l'avons évoqué, les fichiers affichés sont situés dans le répertoire `vue`. De manière plus précise, le fichier contenant la barre latérale de navigation et qui est affiché quelle que soit la situation est nommé `main.html.php`. Tous les autres fichiers ont des noms relatifs aux objets donc ils sont les vues. Ainsi, `phone.html.php` correspond par exemple à la vue pour les téléphones.

Comme cela a été expliqué, toutes les correspondances sont visibles dans le fichier `vue.php` qui contient la classe `Vue`.

4.1.4 Scripts, styles et autres éléments

Feuilles de style Les feuilles de styles de l'application sont stockées dans le dossier `assets/styles`.

Scripts JS Les scripts Javascript utilisés pour l'application sont stockées dans le dossier `assets/scripts`.

Polices L'application utilise avec parcimonie la police Font Awesome pour afficher des éléments graphiques (✓, 👤, ✕,...). Cette police est contenue dans le dossier `assets/fonts`.

Fichiers XSL et XML Ces fichiers utilisés pour la génération du PDF (voir Section 4.4) sont directement disposés dans le dossier `vue`, car directement liés à ces dernières.

4.2 Connexion à la base de données

La connexion à la base de données est réalisée par la classe `DB`. Cette classe offre les fonctionnalités discutées dans cette section.

4.2.1 Connexion et déconnexion

La connexion est réalisée dans le constructeur, en utilisant `mysqli`. En particulier, les lignes suivantes permettent de configurer la connexion à la base de données et sont à adapter à votre propre cas :

```
1 $this->servername = "localhost";
2 $this->username = "root";
3 $this->password = "0000";
4 $this->dbname = "telephonie";
```

En cas d'erreur, cette classe renvoie une erreur pour avertir l'utilisateur que la connexion a échoué. La déconnexion est assurée via la méthode `close`.

4.2.2 Exécution de requête

Pour exécuter des requêtes, deux méthodes sont mises à disposition :

- ⊙ `execute` Cette méthode permet d'exécuter les requêtes une à une.
- ⊙ `executeMulti` Cette méthode permet d'exécuter plusieurs requêtes à la fois.

Des exemples d'appels vous sont fournis ci-dessous :

```
1 // La requête à exécuter
2 $sql = "SELECT * FROM forfait_etranger WHERE is_deleted = FALSE;";
3 // Création de la connexion
4 $db = new DB();
```

```

5 // Exécution de la requête
6 $result = $db->execute($sql);

```

Et avec l'utilisation de `executeMulti` :

```

1 $sql = "DELETE FROM forfait_etrananger WHERE id=2;"
2 $sql = $sql . " SELECT * FROM forfait_etrananger WHERE is_deleted = FALSE;";
3 // $sql contient alors deux requêtes d'affilée.
4 $db = new DB();
5 // Ces requêtes seront correctement traitées par executeMulti.
6 $db->executeMulti($sql);

```

L'idée est ici de répondre au besoin, et de permettre d'appeler `mysqli` pour exécuter des requêtes et procédures partout dans l'application sans se poser des questions sur l'appel (encodage, filtrage de l'erreur,...). En particulier, une erreur est renvoyée si des problèmes sont rencontrés. Cette erreur peut donc être récupérée dans les classes qui en ont besoin.

4.2.3 Traitement des chaînes de caractères

Pour permettre de filtrer les champs rentrés par les utilisateurs et échapper les caractères spéciaux, nous avons réalisé une fonction qui permet de faire cela de manière transparente, il s'agit de `escape_var`. Cette méthode appelle juste `mysqli_real_escape_string`, mais avec un nom plus court, et sans devoir préciser systématiquement quelle connexion utiliser.

Un exemple d'appel vous est proposé ci-dessous :

```

1 $sql = "DELETE FROM forfait_etrananger WHERE nom='". $db->escape_var($el) . "'";

```

On s'assure ainsi d'éviter des intrusions SQL dans notre base de données. Nous avons mis ceci en place sur un certain nombre de requêtes, mais il aurait été judicieux de le déployer à toutes les requêtes prenant des paramètres en provenance de la partie client.

4.2.4 Utilisation de fonctions

Initialement, nous avons hésité entre utilisation de fonctions ou de procédures dans notre application. Si à terme nous avons plus penchés pour des procédures, nous fournissons également dans la classe `DB` une méthode permettant l'exécution de fonctions, il s'agit de `callSQLFunction`.

4.3 | Les requêtes SQL

Notre stratégie a été de créer un maximum de procédures SQL de manière à simplifier les appels ultérieurs. Ainsi les insertions comme les opérations plus délicates utilisent des procédures. Dans le cas où les transactions ne sont pas nécessaires et où une seule valeur de retour est requise, on peut également utiliser une fonction.

Cette section n'a pas pour vocation de présenter toutes les procédures mises en place mais plutôt de présenter la démarche entreprise et quelques requêtes caractéristiques.

4.3.1 Insertion dans la base

Prenons ici l'exemple de l'insertion d'une nouvelle consommation dans la base de données. On crée alors la procédure `addConso` qui permet de réaliser le travail voulu. Grâce à cette procédure nous pouvons également récupérer l'ID de la consommation nouvellement insérée. Il est donc nécessaire de recourir à une transaction afin que l'ID de ligne ajoutée corresponde au dernier ID de la table. Le code SQL pour enregistrer cette procédure est donné ci-dessous :

```

1
2 DROP PROCEDURE IF EXISTS addConso |
3

```

```

4 CREATE PROCEDURE addConso (
5     IN debut DATETIME,
6     IN conso INT(11),
7     IN achat INT(11),
8     INOUT id INT)
9 BEGIN
10     IF NOT EXISTS (SELECT * FROM consommation WHERE idconsommation=id AND
11                     date_debut=debut AND id_achat=achat) THEN
12         START TRANSACTION;
13         /* insert new user */
14         INSERT INTO consommation (date_debut, conso_data, id_achat) VALUES (debut,
15                                         conso, achat);
16         /* get generated id */
17         SET id = (SELECT idconsommation FROM consommation WHERE date_debut=debut
18                     AND conso_data=conso AND id_achat=achat ORDER BY idconsommation DESC
19                     LIMIT 1);
20         COMMIT;
21     END IF;
22 END |

```

L'appel de cette procédure se fait de la manière suivante :

```

1 CALL addConso('20161100', 10, 1, @id); /*consommation de novembre 2016 pour l'achat
dont l'ID est 1*/

```

Il suffit ensuite de récupérer l'ID de la consommation générer de la manière suivante :

```

1 SELECT @id;

```

4.3.2 Une requête plus évoluée

La procédure `getTotalPriceConso` est l'une des plus compliquée car elle permet de calculer le prix total d'une consommation et donc tenir compte de tous les SMS et MMS envoyés, de tous les appels passés, de la consommation de données, de l'endroit vers lequel ont été fait ces opérations et l'heure à laquelle elles ont été faites. On considère ici que si un appel commence pendant une période illimitée alors il est gratuit même s'il se conclut après la fin de la période de gratuité.

Attachons nous déjà à calculer le prix des SMS et MMS émis vers la France et qui dépende donc uniquement du forfait principal. Les opérations à réaliser sont les suivantes :

- 1 Récupérer uniquement les SMS et MMS correspondant à la consommation en question et ceux émis vers la France
- 2 Garder uniquement ceux envoyés en dehors des périodes de gratuités
- 3 Sommer le volume des messages restants
- 4 Comparer ce volume à la limite de SMS autorisés
- 5 Si ce nombre est plus grand, multiplier la différence par le coût d'un SMS hors forfait et l'ajouter au coût total.

Ce qui nous donne la portion de procédure suivante (la France a l'ID 1 dans la liste des pays) :

```

1 /* get the number of sms and mms sent outside every unlimited period to a french
2 phone for the right month*/
3 SET numberFrenchSMS = (SELECT SUM(volume) FROM /*get sum of the remaining volume of
4 sms and mms */
5 (SELECT volume, isInPlageHoraire(date, formule_plage_horaire.plage_horaire) as
6 appartient_plage
7 FROM (
8     SELECT CONCAT('m', idmms) as id, volume, date
9     FROM mms WHERE consommation=consoId AND destination=1
10    UNION ALL /* union des sms (dont l'id est préfixé par s) et des mms (dont
11               l'id est préfixé par m) */

```

```

8      SELECT CONCAT('s',idsms) as id, volume, date
9      FROM sms WHERE consommation=consoId AND destination=1) u,
          formule_plage_horaire
10     WHERE formule_plage_horaire.formule=@formule_id /*récupération des plages
          horaires correspondant à la formule */
11     GROUP BY u.id /* pour chaque sms/mms...*/
12     HAVING SUM(appartient_plage)=0)
13     /* ... on regarde si la date d'envoi est compris dans au moins une plage */
14     temp);
15     IF numberFrenchSMS > @limite_sms THEN
16         SET cost = cost+(numberFrenchSMS-@limit_sms)*@prix_sms;
17     END IF;

```

Calculons maintenant le coût des SMS et MMS émis vers l'étranger. Les étapes sont plus nombreuses :

- 1 Récupérer uniquement les SMS et MMS correspondant à la consommation en question
- 2 Associer les messages à leur forfait étranger parmi ceux associés au forfait principal en fonction de leur destination
- 3 Garder uniquement ceux qui, pour chaque forfait, ne correspondent pas à une période de gratuité
- 4 Sommer le volume des messages restants
- 5 Comparer ce volume à la limite de SMS autorisés (ce qui dépend du forfait étranger considéré)
- 6 Si ce nombre est plus grand, multiplier la différence par le coût d'un SMS hors forfait et l'ajouter au coût total.

Pour réaliser cette opération, nous effectuons une première requête semblable à la précédente. Puis, à l'aide d'un **CURSOR** nous parcourons les résultats afin de comparer la somme de message de chaque forfait à sa limite et le cas échéant ajouter le prix au prix total :

```

1 BEGIN
2     DECLARE done INT DEFAULT FALSE;
3     DECLARE enumber_sms INT;
4     DECLARE elimite_sms INT;
5     DECLARE eprix_hors_forfait_sms FLOAT;
6     DECLARE smscursor CURSOR FOR (SELECT SUM(volume),limite_sms,
          prix_hors_forfait_sms FROM
7         (SELECT sms.volume, limite_sms, prix_hors_forfait_sms,
          isInPlageHoraire(sms.date, forfait_etrange_plage_horaire.plage)
          AS appartient_plage, forfait_etrange.id as id_forfait
8         FROM formule_forfait_etrange,
          forfait_etrange,
9         (SELECT CONCAT('m',idmms) as id, volume, date, destination FROM
          mms WHERE consommation=consoId
10        UNION ALL
11        SELECT CONCAT('s',idsms) as id, volume, date, destination FROM
          sms WHERE consommation=consoId)
12        sms,
13        zone_geographique_pays,
14        forfait_etrange_plage_horaire
15        WHERE (formule_forfait_etrange.formule = @formule_id
16        AND formule_forfait_etrange.forfait_etrange =
          forfait_etrange.id
17        AND zone_geographique_pays.pays = sms.destination
18        AND forfait_etrange.zone = zone_geographique_pays.
          zone_geographique
19        AND forfait_etrange_plage_horaire.forfait= forfait_etrange.
          id)
20        GROUP BY sms.id
21        HAVING SUM(appartient_plage)=0)
22        limited_sms
23        GROUP BY id_forfait);

```



```

25 DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
26
27 OPEN smscursor;
28 smsLoop : LOOP
29     FETCH smscursor INTO enumber_sms, elimite_sms, eprix_hors_forfait_sms
30     ;
31     IF done THEN
32         LEAVE smsLoop;
33     END IF;
34     IF enumber_sms > elimite_sms THEN
35         SET cost = cost + (enumber_sms - elimite_sms)*
36         eprix_hors_forfait_sms;
37     END IF;
38 END LOOP;
39
40 CLOSE smscursor;
41 END;

```

La totalité de la procédure est consultable dans le fichier `consommation_procedures.sql`.

4.4 La génération du PDF de consommation détaillée

Pour pouvoir générer le PDF nous avons besoins de plusieurs choses :

- 1 Récupérer les informations dans la base de données
- 2 Générer un fichier XML avec ces informations
- 3 Associer ce fichier XML à une feuille de style XSLT qui permet de transformer le document XML en document XSL:FO.
- 4 Générer le fichier avec FOP
- 5 Rediriger l'utilisateur vers le PDF.

4.4.1 Récupérer la consommation dans la base de données

Les données doivent être triées selon l'ordre chronologique. Cette opération est réalisée au moyen de la requête SQL suivante :

```

1 SELECT date, volume, nom as destination, zone FROM sms JOIN (
2     SELECT pays.id, pays.nom as nom, zone_geographique.nom as zone FROM pays,
3     zone_geographique, zone_geographique_pays
4     WHERE pays.id=zone_geographique_pays.pays AND zone_geographique.id=
5     zone_geographique_pays.zone_geographique
6 )
7 pays ON (sms.destination=pays.id) WHERE consommation=consoId ORDER BY date ASC;

```

4.4.2 Génération du fichier XML

A partir des informations précédemment récupérée, on vient remplir un fichier XML avec le script PHP partiel suivant :

```

1 $result = $db->execute($sql);
2 if ($result->num_rows > 0) {
3     while($row = $result->fetch_assoc()) {
4         $date = new DateTime($row['date']);
5         echo "<sms>\n";
6         echo "<date>". $date->format('d/m \à H\hi' ). "</date>\n";
7         echo "<volume>". $row['volume']. "</volume>\n";
8         echo "<destination>". $row['destination']. "</destination>\n";

```

```

9         echo "<zone>".$row['zone']."</zone>\n";
10        echo "</sms>\n";

```

Ce qui donne le fichier XML suivant :

```

1  <?xml version="1.0" encoding="utf-8"?><?xml-stylesheet type="text/xsl" href="conso.
    xsl"?><conso>
2  <generationdate>22/01 à 21:06:09</generationdate><smslist>
3  <sms>
4  <date>02/11 à 12h42</date>
5  <volume>6</volume>
6  <destination>Rwanda</destination>
7  <zone>Reste du monde</zone>
8  </sms>
9  ...
10 </smslist>
11 <mmslist>
12 <mms>
13 <date>01/11 à 18h38</date>
14 <volume>3</volume>
15 <destination>Argentine</destination>
16 <zone>Amérique du Sud</zone>
17 </mms>
18 ...
19 </mmslist>
20 <appellist>
21 <appel>
22 <date>01/11 à 12h49</date>
23 <duree>00:28:57</duree>
24 <destination>Arabie Saoudite</destination>
25 <zone>Reste du monde</zone>
26 </appel>
27 ...
28 </appellist>
29 </conso>

```

4.4.3 La transformation en PDF

La feuille de style XSLT ne sera pas expliquée ici car ce n'est pas le sujet principal. Néanmoins, le lecteur curieux pourra la trouver dans le fichier `conso.xsl`. Il s'agit de venir peupler trois tableaux XSL :FO avec les données XML.

On transforme ensuite cette combinaison XML et XSLT en PDF grâce à l'utilitaire Apache FOP qui est écrit en JAVA. La génération est lancée par le script suivant (lequel est appelé par PHP).

```

1 cd fop/build
2 java -jar fop.jar -xml ../../conso.xml -xsl ../../conso.xsl -pdf ../../conso.pdf
3 cd ../../

```

Il suffit ensuite de rediriger la page vers le fichier PDF.

```

1 header('Location: conso.pdf');

```

5 • Améliorations possibles

Afin d'avoir un fonctionnement plus proche d'un vrai site de téléphoniste, les améliorations suivantes seraient envisageables :

- ⊙ *Mécanisme de commande* Le mécanisme de commande actuel est assez rudimentaire. Il serait intéressant que ce processus se fasse par des formulaires successifs qui enregistrent les choix des clients au fur et à mesure et propose un résumé à la fin à valider. Ce processus permettrait également de créer des achats avec formule et téléphones groupés, ce qui n'est aujourd'hui pas possible (à moins d'écrire la requête à la main!).
- ⊙ *Recherches avancées* Si nous avons mis pour l'exemple un mécanisme de recherche avancée pour la table des abonnements, il serait intéressant de généraliser (et améliorer) ce fonctionnement aux autres tables, avec des requêtes SQL créées à la volée pour remplir les conditions attendues.
- ⊙ *Tri dans les tables* Un autre élément d'ergonomie intéressant aurait été d'offrir la possibilité aux utilisateur de réaliser des tris dans les tables (pas prix, par promotion,...). Ceci n'a pas été mis en place dans les interfaces, bien que cela soit mis en place pour la génération du PDF de facture, qui montre donc un exemple de réalisation que nous n'avons pas eu le temps de généraliser à toute l'application.
- ⊙ *Dynamisme* De manière générale, l'interface (bien qu'utilisant du Javascript), reste assez statique, un travail pourrait donc être à réaliser dans ce sens. L'aspect graphique et ergonomique pourrait mériter plus de travail, nous avons cependant préféré réaliser les fondamentaux nécessaires à pouvoir utiliser l'application dans le périmètre fixé initialement.
- ⊙ *Sécurité et protection de la vie privée* La sécurité n'était pas notre priorité et de ce fait, beaucoup de failles sont certainement présentes. On citera notamment le non hashage des mots de passe et la possibilité d'avoir accès aux consommations de n'importe qui via la génération du détail des consommations.

5 • Conclusion

En conclusion, ce travail nous a permis d'appliquer une partie des nouveautés abordées lors de ce module. En particulier, notre application a essayé de tirer le meilleur parti de l'utilisation des procédures. Ainsi, ces dernières nous ont permis de stocker la plupart de nos requêtes en base de données, et ainsi de faciliter leur appel dans le code.

Pour le code, le choix a été d'utiliser une technologie vue lors du S5 de Centrale Lyon, à savoir d'utiliser le langage PHP en se connectant à une base de données SQL en utilisant `mysqli`.

Outre les aspects SQL, ce projet fut également une occasion d'implémenter une utilisation de feuilles de XSL pour réutiliser des données en provenance d'une base de données MySQL et de les mettre en forme (ici sous forme de PDF).

Dans un dernier temps, ce projet nous a permis d'essayer de voir comment l'utilisation de SQL pouvait s'intégrer dans la mise en place d'une architecture MVC pour une application/plateforme web.

Ce projet fut donc une occasion de tester plusieurs technologies et principes et de voir leurs interactions. Si certains aspects de la plateforme auraient pu être plus développés, nous avons ici pris le parti de tester des solutions diverses, afin de voir les possibilités des langages et leurs relations.