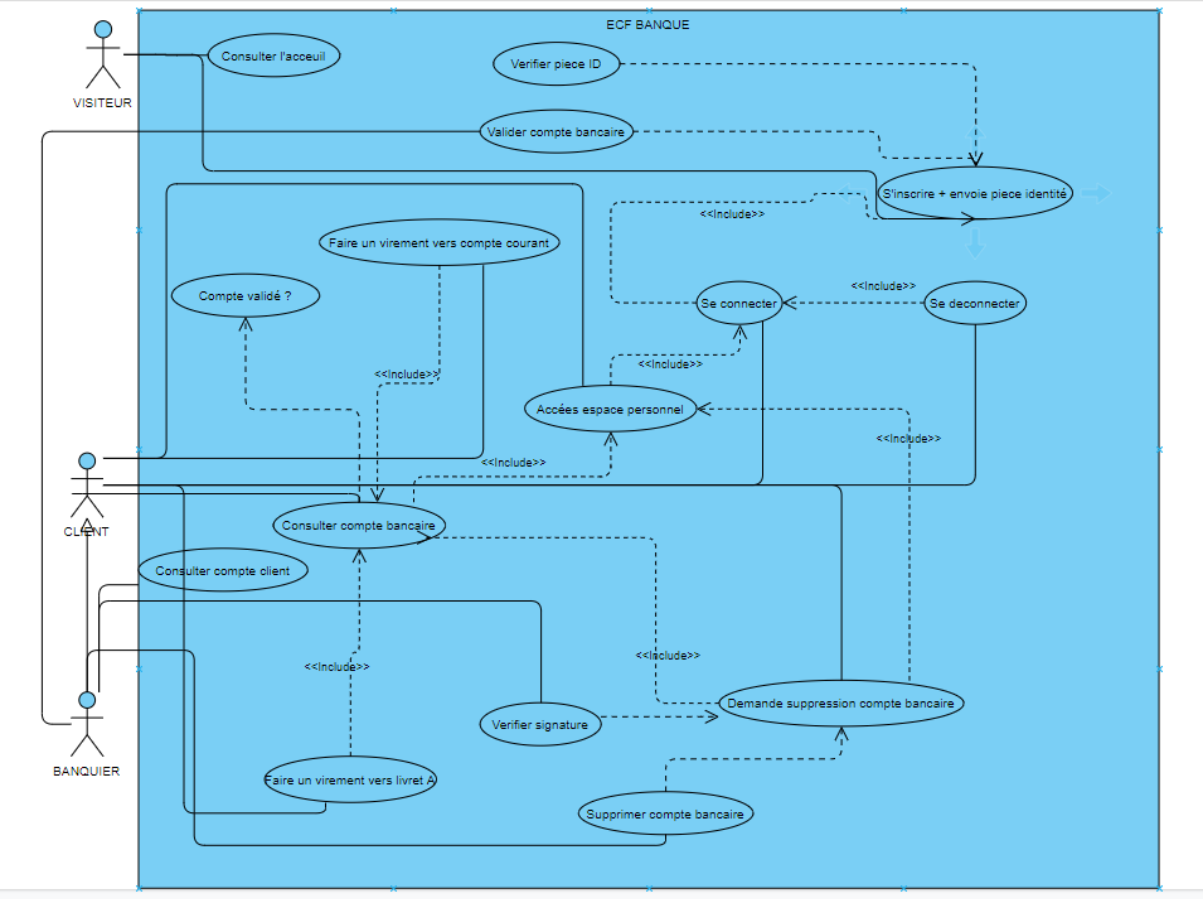
DOCUMENT QUESTIONS ET REFLEXION

1 – Le choix des technologies

Pour le Front j'ai choisi : Html, css, javascript, bootstrap.
Pour le Back j'ai choisi : Php 7.2 avec le framwork Symfony 5.
Pour la Base de données j'ai choisi : Mysql + Doctrine.
Mon IDE : PHPSTORM.
SERVER : WAMP.
Logiciel d'exploitation : WINDOWS 10.

2 – Les UML

J'ai maquetté un USE CASE avec VISUAL PARADIGME.

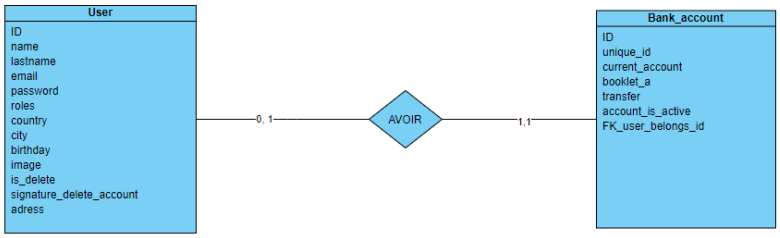


J'ai maquetté un diagramme de classe avec VISUAL PARADIGME.

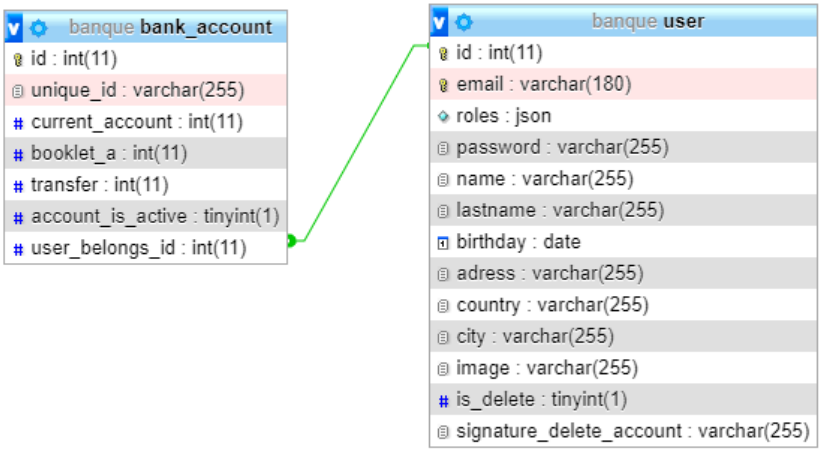
- Je me suis demander comment un admin pourrais validé un compte ?
  Pour cela j'ai établie une variable dans ma table Bank_account que j'ai nomé « account_is_active », qui sera un booléan, cela vraudra soit true ou false, grace a cela je vais pouvoir établir une condition qui va afficher un compte bancaire que si cet variable est sur true.

Comment un utilisateur pourrais demandé une demande de suppression de compte ?

- J'ai établie une variable dans ma table User que j'ai nomé « is_delete » qui sera un boolean et une autre que j'ai nomé « signature_delete_account » qui prendra une image de la signature du client.



Le Modèle physique de données.

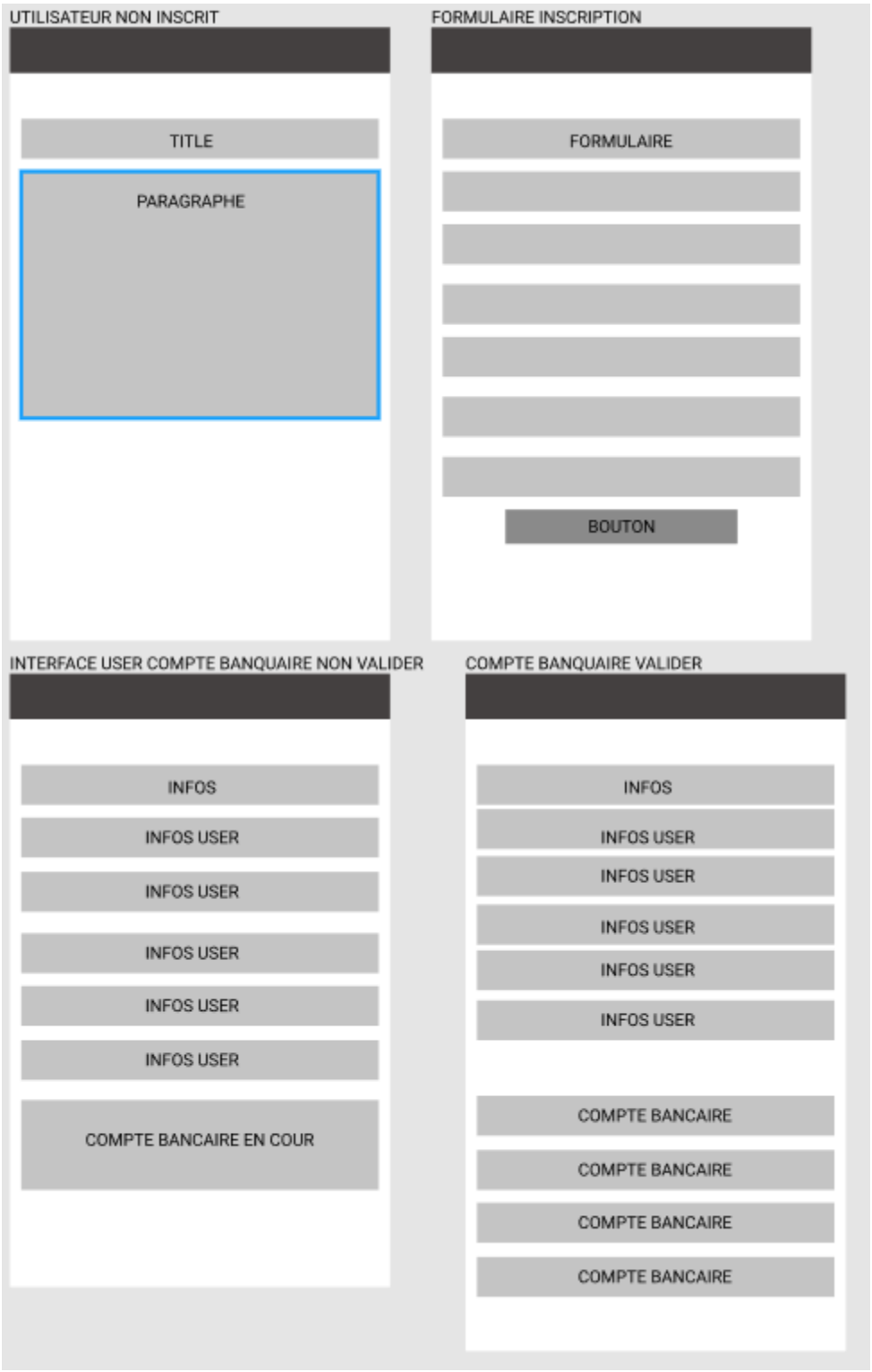


3 – Les maquettes

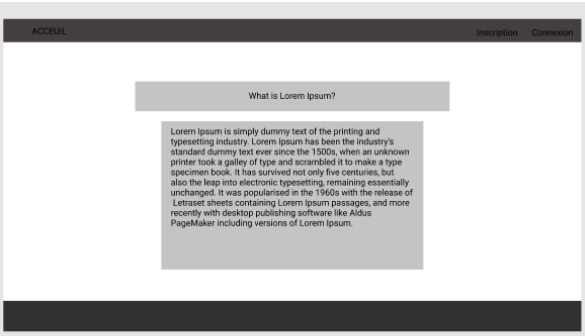
J’ai fait le choix de réalisé le zoning, wireframe avec figma et le mock-up avec adobe xd.

ZONING

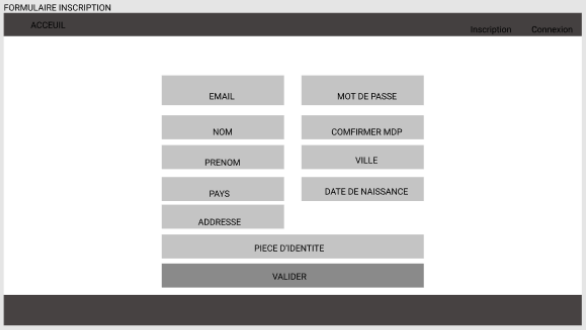
ZONING MOBILE


UTILISATEUR NON INSCRIT
TITLE
PARAGRAPHE
FORMULAIRE INSCRIPTION
FORMULAIRE
BOUTON
INTERFACE USER COMPTE BANQUAIRE NON VALIDER
INFOS
INFOS USER
INFOS USER
INFOS USER
INFOS USER
INFOS USER
COMPTE BANCAIRE EN COUR
COMPTE BANQUAIRE VALIDER
INFOS
INFOS USER
INFOS USER
INFOS USER
INFOS USER
INFOS USER
COMPTE BANCAIRE
COMPTE BANCAIRE
COMPTE BANCAIRE
COMPTE BANCAIRE

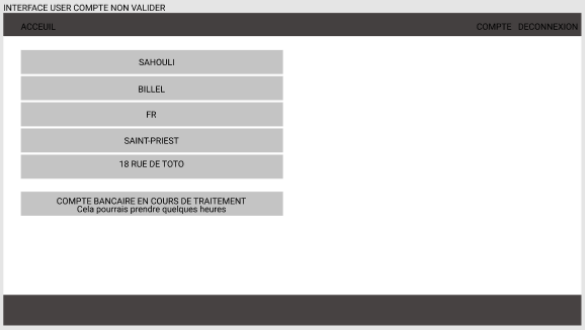
WIREFRAME



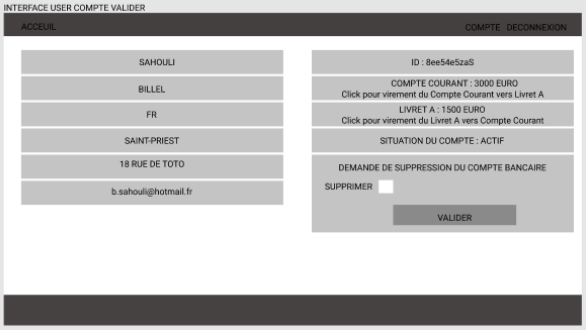
FORMULAIRE INSCRIPTION



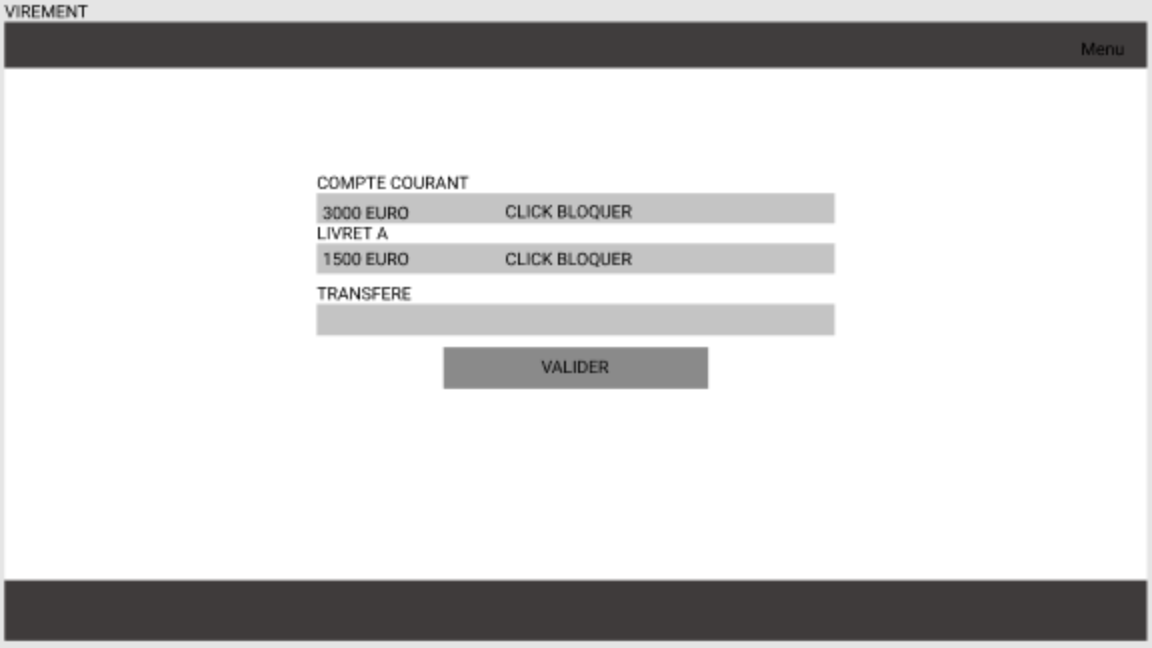
INTERFACE USER COMPTE NON VALIDER

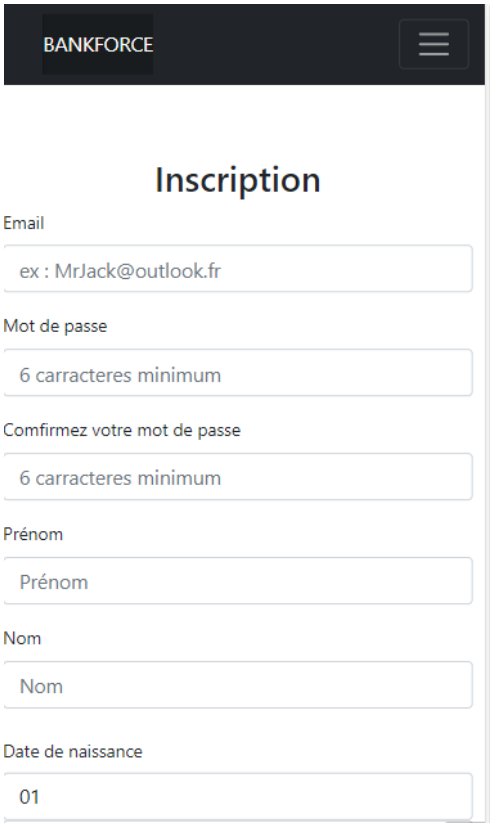


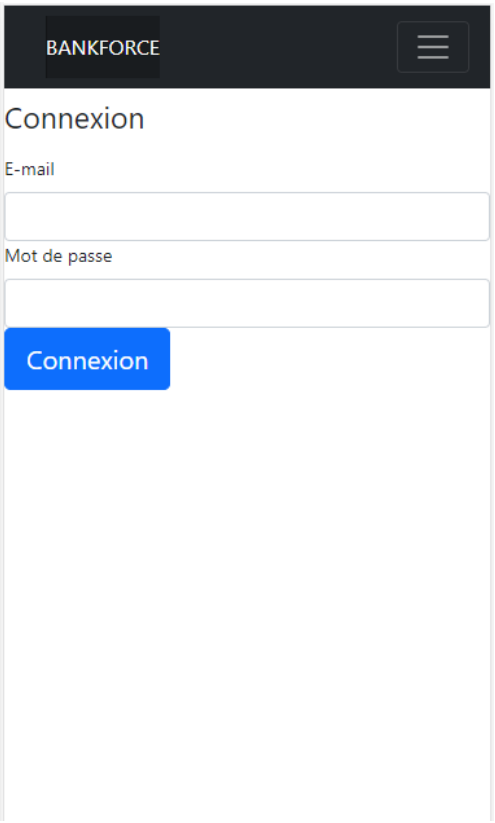
INTERFACE USER COMPTE VALIDER



VIREMENT







4 – Le choix de l'architecture et configuration

J'ai fait le choix de suivre une architecture MVC (Modèle Vue Contrôleur).
Le principe est simple, dans mon cas j'utilise Doctrine, Doctrine vas générer les entités de mes tables qui vont représentez le Modèle. Le contrôleur lui va être le chef d'orchestre il vas non seulement se servir du Modèle pour enregistrer et envoyer de la data dans ma base de données mais il va aussi renvoyer des données a la Vue, et la Vue vas se charger d'afficher les données à l'utilisateur.

Par la suite dans le fichier «.env » je peux passer en environnement de dev ou de prod.

```
###> symfony/framework-bundle ###
APP_ENV=dev
APP_SECRET=842f92e4b05a6db98df4fc2c7a95bbdb
```

Toujours dans le même fichier je peux créer ma base de données et la configurer en mettant le dbUser le mot de passe et le nom de la base de données que je veux lui donner.

```
DATABASE_URL="mysql://root:@127.0.0.1:3306/banque?serverVersion=5.7"
```

5 – Les bonnes pratiques de sécurité

Dans le security.yaml il y a la gestion du hachage du mot de passe, il va prendre le meilleur type de hachage possible, dans mon cas il a sélection le hachage en « argon ».

```yaml
security:
    encoders:
        App\Entity\User:
            algorithm: auto
```

Là je peux gérer aisément la sécurité des connexions, lorsqu'un utilisateur s'inscrit son rôle est ROLE_USER, ainsi pour déclarer qu'un user est un admin je doit le modifier dans ma table et lui insérez le ROLE_ADMIN.

```yaml
access_control:
    - { path: ^/admin, roles: ROLE_ADMIN }
    - { path: ^/compte, roles: ROLE_USER }
```

	id	email	roles	password	name	lastnam
1	4	b.sahouli@hotmail.fr	["ROLE_ADMIN"]	$argon2id$v=19$m=65536,t=4,p=1$MzN1UG1ORFU4Tk…	Billel	Sahouli
2	5	edede@hotmail.fr	[]	$argon2id$v=19$m=65536,t=4,p=1$UVJBVHYuc1p0Mm…	Billel	Sahouli
3	6	dddd@hotmail.fr	[]	$argon2id$v=19$m=65536,t=4,p=1$OUpLTVRoOFdCVH…	Billel	Sahouli
4	11	testDate@hotmail.fr	[]	$argon2id$v=19$m=65536,t=4,p=1$VzJJakhJdkhRN2…	Billel	Sahouli

Un exemple de mes routes sécurisé.

```php
/**
 * @Route("/compte/virement-compte-courant/{id}", name="account_current")
 */
public function transferCurrentAccount($id,Request $request): Response
```

```php
/**
 * @Route("/admin/delete/{id}", name="admin_delete_account")
 */
```