

PTUT : Application gestion de sondage

1. Le besoin

1.1. Objectif principal

Le premier objectif du projet est de récupérer des réponses de sondage (issues de l'application sur la tablette), de les stocker sur un serveur, de les traiter, et les mettre en forme afin d'exploiter ces résultats.

1.2. Objectif secondaire

La création de sondage sur le site web et l'application Android, par un utilisateur connecté au site/API.

La création de sondage implique de renseigner un titre et une description. Chaque sondage sera composé d'une ou plusieurs pages, qui contiendront les questions, et les propositions.

Une question sera composée d'un titre, d'un type de proposition (liste ci-dessous), et sera liée à plusieurs propositions :

- À choix multiple,
- Des cases à cocher,
- Une échelle linéaire (de 1 à $\pm n$),

Les utilisateurs normaux peuvent administrer les sondages qu'ils ont créé :

- Edition textuelle de questions/propositions (corrections orthographiques uniquement),
- Voir et télécharger les résultats/graphiques du sondage.

Le (ou les) administrateur(s) possèdent les mêmes droits que les utilisateurs normaux, mais sur tous les sondages existants.

2. Fonctionnalités

2.1. Récupération des données de la tablette

Après avoir répondu à un questionnaire sur la tablette, on assume que les données issues de ce questionnaire seront envoyées à un serveur, et ensuite stockées en mémoire.

2.2. Mise en forme des données, sous forme de tableaux et/ou de graphiques

La mise en forme de données des tableaux se fera directement en HTML. Pour le rendu des graphiques, une librairie tierce sera utilisée.

2.3. Calcul de moyennes pour les données des questions quantitatives

Pour chaque question d'un sondage, on affichera un diagramme en barres des résultats.

Pour les questions quantitatives (où l'utilisateur doit rentrer/choisir un chiffre pour répondre), on affichera la moyenne des réponses.

2.4. Création d'un fichier Excel des résultats tableau et graphique

La création de fichier "Excel" se fera pour un, ou plusieurs sondages (un fichier par sondage). Pour ce fichier, on utilisera une librairie permettant de générer un fichier respectant le format OpenDocument.

2.5. Création d'une archive des enquêtes dans un répertoire

Le fichier "Excel" sera généré la première fois qu'un utilisateur souhaite voir les résultats de son sondage.

Ce fichier sera ensuite gardé en cache, et un bouton permettra à l'utilisateur régénérer ce fichier.

L'utilisateur aura la possibilité de télécharger une archive au format .zip, qui contiendra les fichiers Excel d'un ou plusieurs sondages à la fois.

3. Technologies retenues

3.1. Communication

La communication entre le serveur et l'application Android se fera en JSON :

- Serveur => Transmission des sondages => Application Android,
- Application Android => Transmission des réponses pour un sondage => Serveur.

Pour le site web, la communication restera la même, mais sans utiliser de JSON.

Un exemple des routes qui seront utilisées se trouvent ci-dessous, il est probable que d'autres routes seront rajoutées par la suite.

Méthode	URL	Utilité
Routes utilisées par le site web		
GET	/polls	Affiche les sondages (sous forme pagination).
GET	/polls/create	Retourne le formulaire de création de sondage.
GET	/polls/{poll}	Affiche le sondage {poll} + questions + propositions.
GET	/polls/{poll}/results	Retourne les résultats du sondage {poll}.
GET	/user/polls	Retourne les sondages créés par l'utilisateur.
Routes utilisées par l'API		
GET	/api/polls	Retourne les sondages sous le format JSON.
GET	/api/polls/{poll}	Retourne le sondage {poll} + questions + propositions, sous le format JSON.
GET	/api/polls/{poll}/results	Retourne les résultats du sondage, sous le format JSON.
GET	/api/user/polls	Retourne les sondages créés par l'utilisateur {user} sous le format JSON.
Routes utilisées par le site web et l'API		
POST	/polls	Crée un nouveau sondage
POST	/polls/:poll_id	Sauvegarde les réponses envoyées par un

		utilisateur
--	--	-------------

3.2. Laravel

Laravel est un framework PHP, et sera utilisé pour la création du site web ainsi que l'API JSON.

Il comprend la gestion de routes, modèles, vérifications des données envoyées par un formulaire, et de rendu grâce au moteur de template Blade.

3.3. Affichage des graphiques sur le site web : Google chart

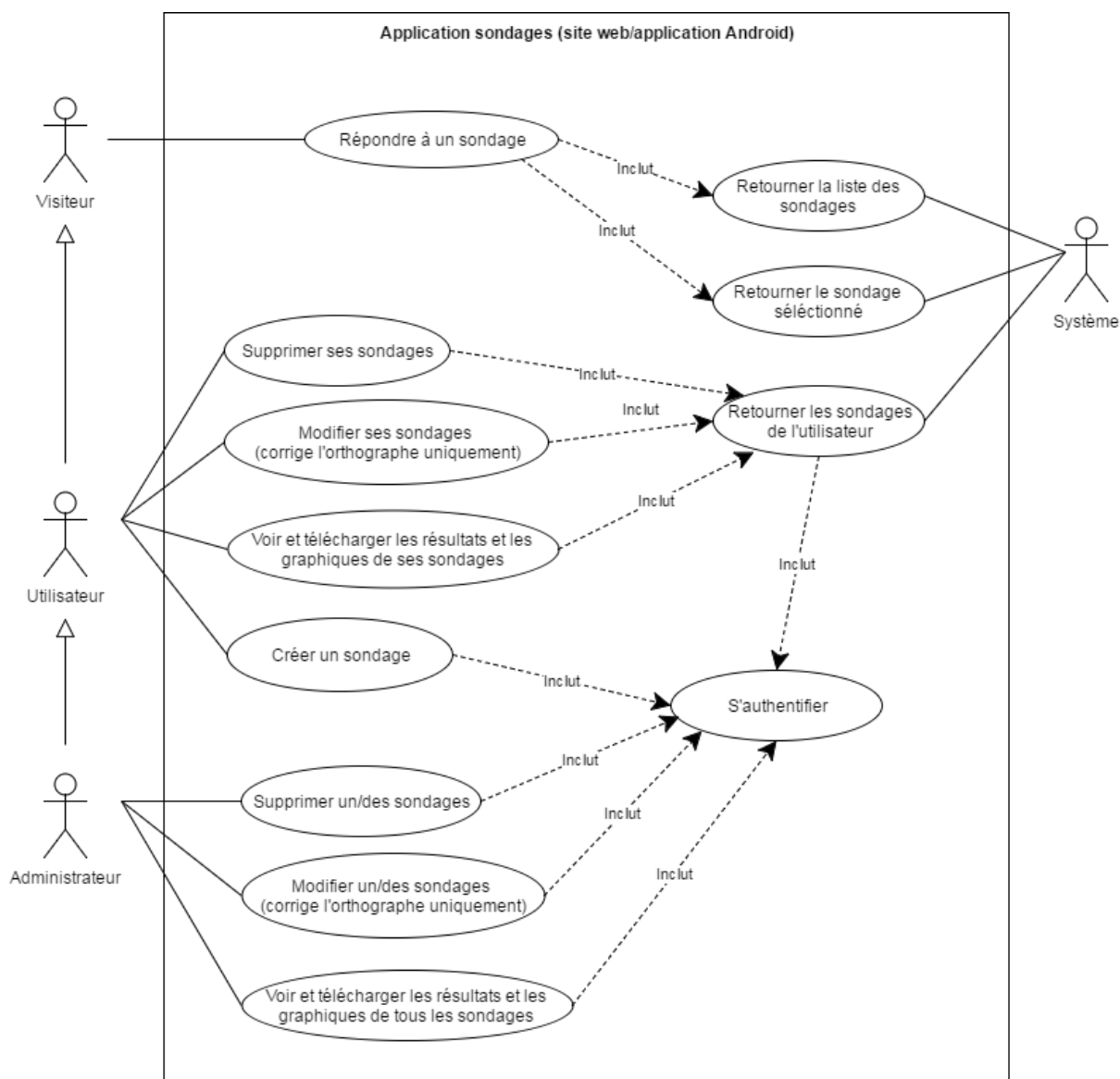
Google chart permet la génération de graphiques à plusieurs entrées et l'affichage des données sous forme de tableaux ainsi qu'un export CSV des données et PNG du graphique.

3.4. Génération de document OpenDocument

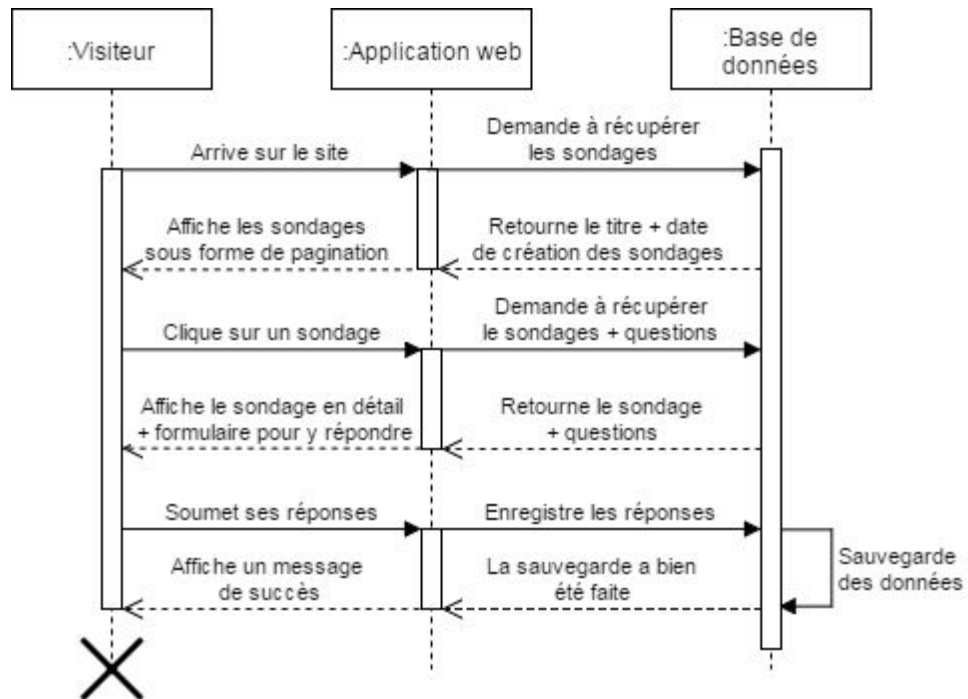
La génération des graphiques et tableaux se fera grâce à une librairie PHP appelée PHPWord. Elle intègre la génération de graphiques et la sauvegarde de fichier au format OpenDocument.

4. Diagrammes

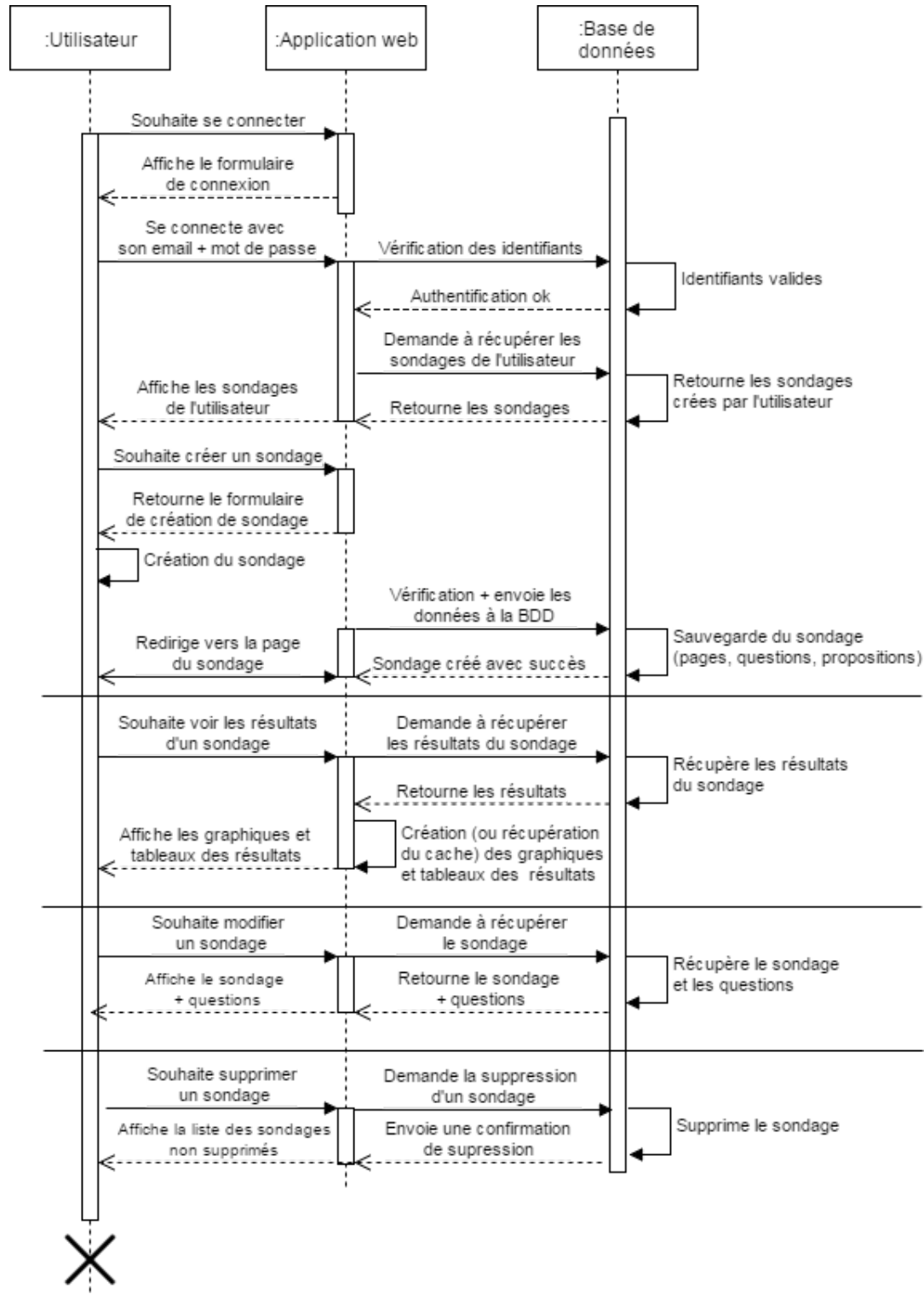
4.1. Diagramme de cas d'utilisation



4.2. Diagramme de séquence (Visiteur)



4.3. Diagramme de séquence (Utilisateur)



4.4. Diagramme de séquence (Administrateur)

Le diagramme de séquence de l'administrateur est le même que l'utilisateur, mais il a la possibilité de, voir, supprimer, modifier tous les sondages.

4.5. Diagramme de classe

