



Outil de gestion d'évènements
pour les responsables qui se veulent organisés

Aïolah VAÏTI, Dylan THAO et Lucas KERDONCUFF

Document créé le 15 janvier 2020
Dernière modification le 2 mars 2020

SOMMAIRE

HISTORIQUE DE MODIFICATIONS	3
INTRODUCTION	4
ARCHITECTURE LOGICIELLE	5
Partie BACK-END	5
L'API	6
Partie FRONT-END	6
DESCRIPTION DU PRODUIT	7
EXIGENCES	7
CAS D'UTILISATION	9
SPÉCIFICATION DÉTAILLÉE	10
MAQUETTES	16

HISTORIQUE DES MODIFICATIONS

Date	Nom	Action
15 janvier 2020	Lucas	Création du document
22 janvier 2020	Tous	Rédaction des exigences
3 février 2020	Tous	On complète le document
26 février 2020	Aïolah	Modifications suite aux commentaires de Mme Trouilhet, ajout des principales fonctionnalités dans les spécifications détaillées
2 mars 2020	Dylan	Refonte du document: mise en page et ajouts, design des maquettes
	Aïolah	Complétion des spécifications détaillées, mise en page
	Lucas	Complétion des spécifications détaillées, création des diagrammes de séquence

INTRODUCTION

Nous proposons une application de planning événementiel ayant pour objectif de faciliter l'organisation d'événements tel que des anniversaires, des mariages ou autre.

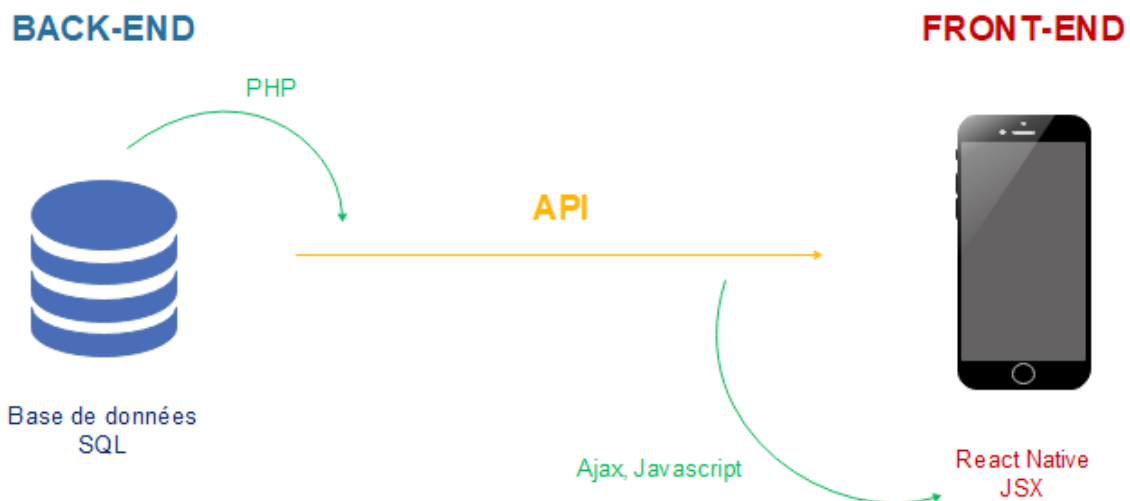
Il est pas toujours aisé de planifier correctement de A à Z un événement. C'est pourquoi nous proposons une application capable de guider un utilisateur durant les différentes étapes de sa création. Krevent vous permettra de jongler habilement entre un agenda de planification, la gestion des invités et participants, un tableau d'attribution de tâches et la gestion des documents administratifs.

A chaque invité sera attribué un rôle entre "Invité" et "Organisateur". Le rôle d'invité permet d'avoir accès à une fiche récapitulative de la soirée, regroupant les informations nécessaires telles que la date, le planning, la liste des invités, le plan de table, le menu etc.. Tandis que les organisateurs, quant à eux, disposeront de l'intégralité des fonctionnalités de l'application. Ils auront donc la possibilité de contacter les invités depuis leurs contacts, de déposer et de consulter des documents administratifs dans un lieu de dépôt commun. De plus, de nombreux outils seront mis à disposition pour faciliter la création d'une liste de courses, d'une liste de choses à faire ou d'un plan de table.

Nos utilisateurs seront continuellement avertis des différents changements concernant l'événement et pourront garder un oeil sur le budget fixé au préalable ainsi que l'évolution de la liste des invités.

Une fois l'événement terminé, l'application sera encore disponible afin que chacun puisse partager et télécharger des photos et vidéos.

ARCHITECTURE LOGICIELLE



Nous avons choisi de réaliser notre projet sous la forme d'une application mobile car celle-ci possède de nombreux avantages comparé à une application web. Une application mobile est en effet plus pratique puisqu'il n'y a pas besoin d'un ordinateur pour s'en servir. Étant donné que notre service est un outil de gestion d'évènements, les utilisateurs seront certainement souvent en déplacement pendant qu'ils se serviront de l'application. Par exemple, au moment de faire leurs courses utiles à l'évènement, ils se serviront de leur liste de course créée sur l'application. Pour de simples raisons pratiques, il est préférable qu'ils puissent utiliser leur téléphone portable plutôt que leur ordinateur à ce moment-là.

Réaliser une application mobile implique de créer une API grâce à laquelle nous pourrions récupérer les informations stockées en base de données. Nous avons ainsi séparé notre architecture logicielle en 3 parties :

1. Partie BACK-END

Les données de l'application seront stockées dans une base de données relationnelle SQL. Celle-ci permet en effet de distinguer chaque entité et de bien modéliser les relations entre chacune d'entre elles.

2. L'API

Les données seront affichées en PHP car il est possible d'utiliser la notion de programmation objet tout en interagissant avec la base de données. Cela nous fera gagner du temps de développement : la plupart des composants de l'application seront modélisés sous forme de classes. En utilisant le PHP, nous pourrons encoder les données au format JSON pour pouvoir ensuite les afficher.

Nous avons choisi d'allier des requêtes Ajax à Javascript pour récupérer les données côté Front.

3. Partie FRONT-END

Nous avons décidé d'utiliser React Native pour la partie front-end pour son adaptabilité, que ce soit pour des mobiles ou bien des tablettes. En effet, React Native reprend les composants natifs à la fois d'iOS et d'Android.

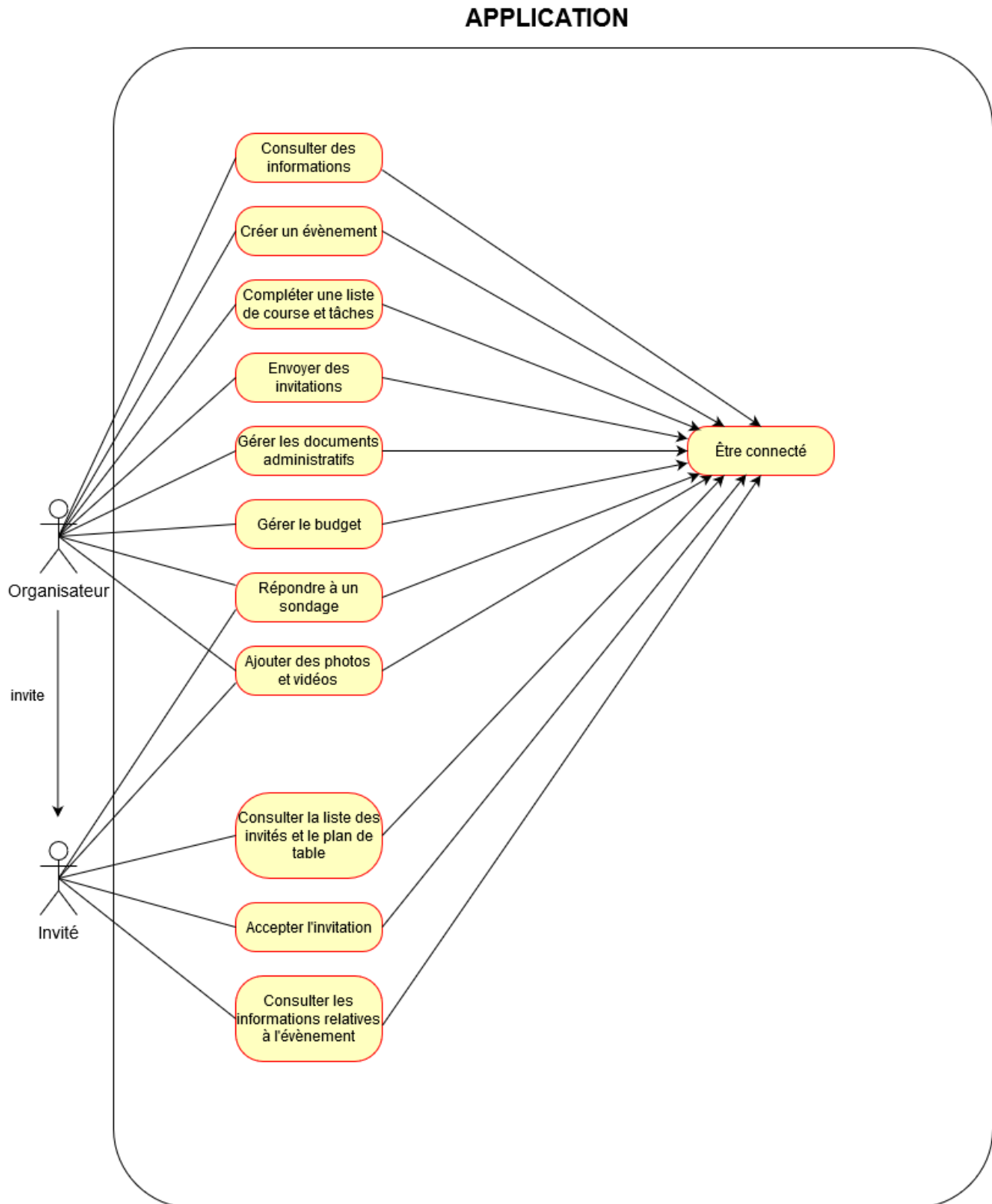
DESCRIPTION DU PRODUIT

- EXIGENCES

NIVEAU DE NÉCESSITÉ	EXIGENCE
MUST	Toutes les informations doivent être stockées dans une base de données
	L'application doit permettre de créer des évènements
	L'application doit permettre de créer un compte
	Un utilisateur doit être connecté pour créer un évènement
	Un organisateur doit pouvoir inviter une personne à son évènement
	Un invité doit être connecté pour consulter toutes les informations d'un évènement auquel il est invité
	Un organisateur doit pouvoir compléter une liste de course, une liste de tâches et un programme
SHOULD	Un organisateur devrait pouvoir attribuer des tâches à un autre organisateur/invité
	Un organisateur devrait pouvoir fixer/consulter le budget
	Un organisateur devrait pouvoir donner une date limite à une tâche
	Un organisateur devrait pouvoir déposer des fichiers importants (réservation etc..)

COULD	L'application pourrait alerter l'organisateur à l'approche d'une date butoir
	L'application pourrait alerter l'organisateur si une tâche lui est attribuée
	L'application pourrait alerter l'organisateur s'il dépasse le budget
	Un organisateur pourrait écrire le menu de l'événement
	Un organisateur pourrait établir le plan de table
	Les organisateurs et invités pourraient déposer des photos/vidéos dans un lieu de dépôt appelé album photo
WON'T	Un organisateur pourrait potentiellement savoir si tous les invités ont payé
	Un organisateur pourrait potentiellement créer un sondage
	Les organisateurs et les invités pourraient potentiellement pouvoir répondre à un sondage
	Un invité pourrait potentiellement contacter les organisateurs et inversement
	Un organisateur pourrait potentiellement créer une liste des prestataires de l'événement
	Un organisateur pourrait potentiellement savoir si un prestataire a été payé

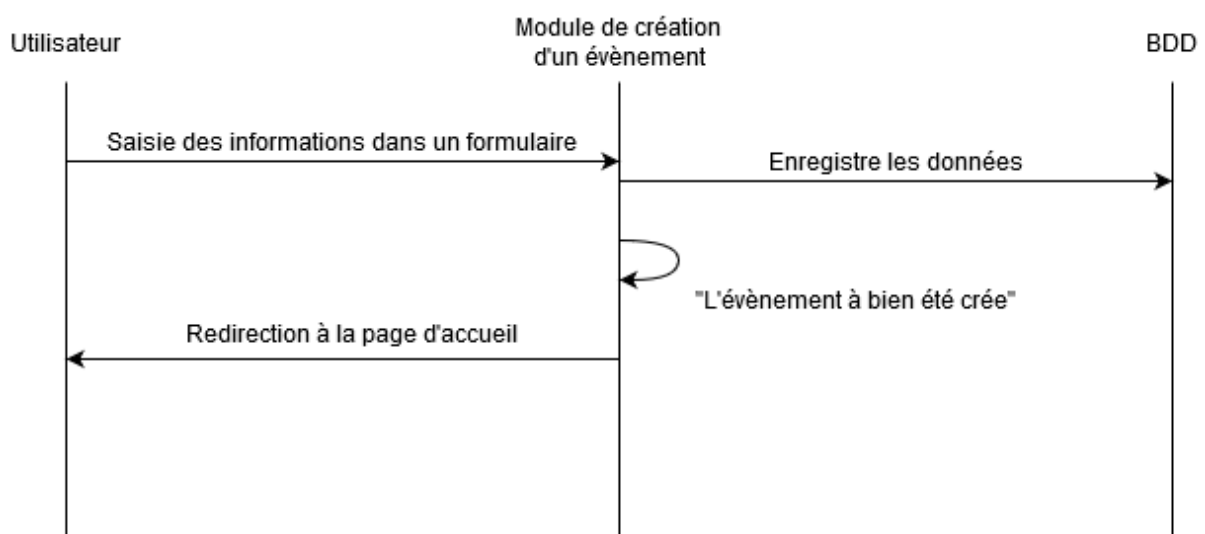
- CAS D'UTILISATION



SPÉCIFICATION DÉTAILLÉE

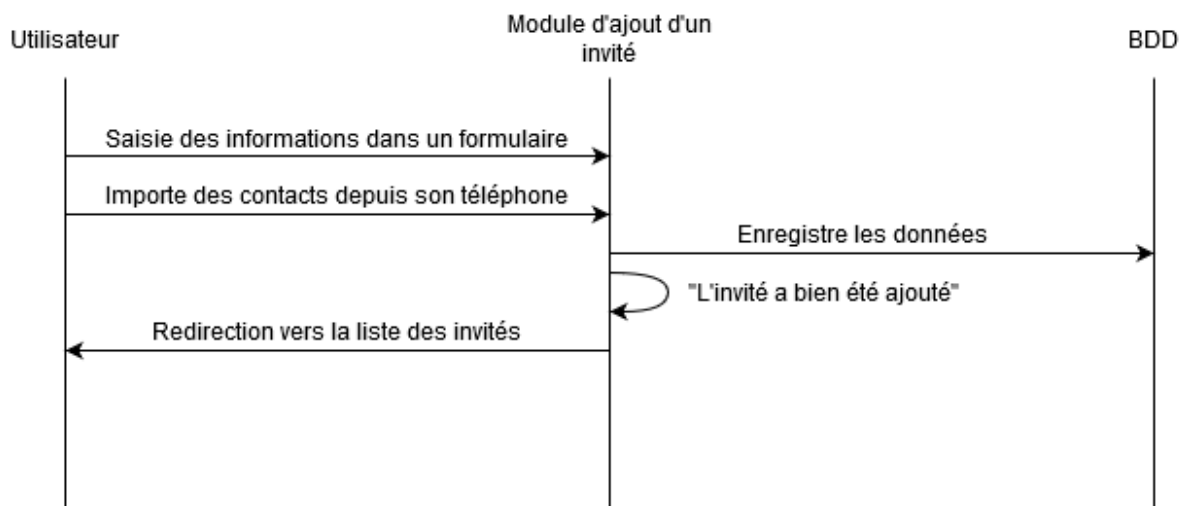
1. Création d'un évènement

Traitement	
Rôle	Enregistre les données propres à un évènement
Paramètres	Nom, lieu, date, heures de début et de fin, type de l'évènement
Pré-condition	L'utilisateur doit être connecté
Postcondition	L'évènement est ajouté en base de données
Tâche	L'utilisateur saisit dans un formulaire les informations relatives à l'évènement puis appuie sur le bouton "Créer"



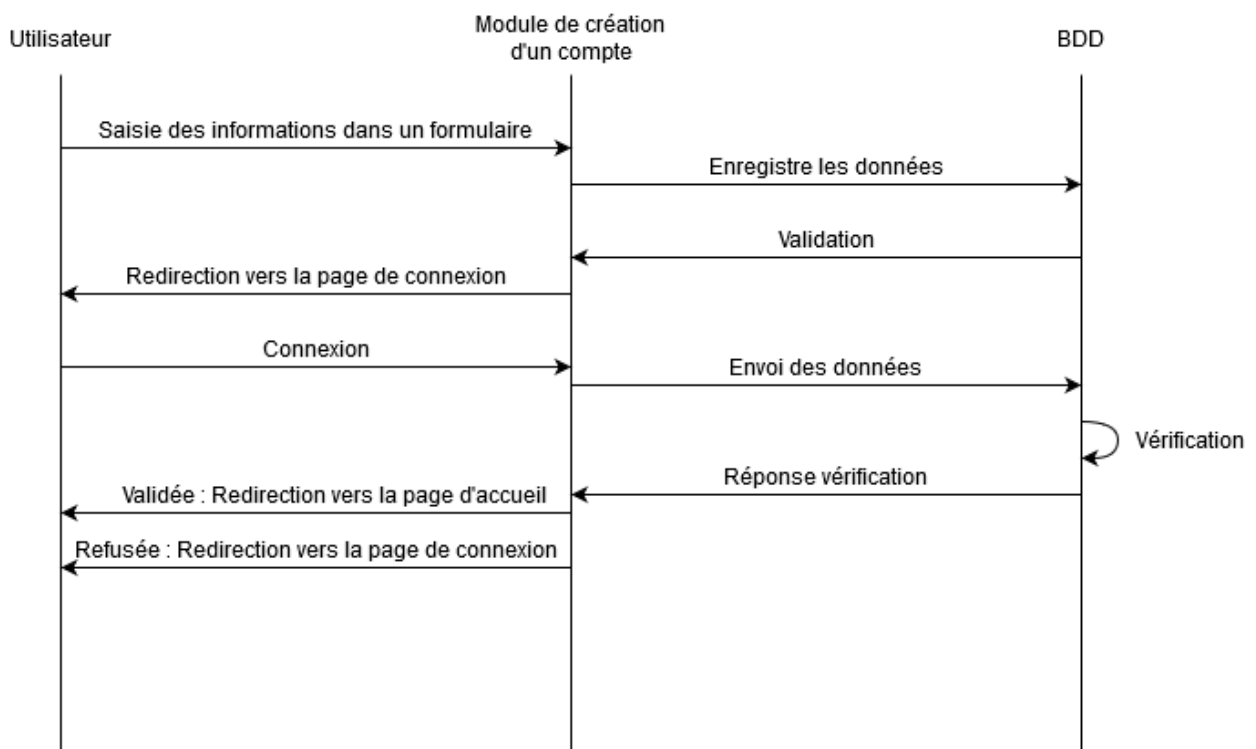
2. Ajout d'un invité

Traitement	
Rôle	Ajoute un invité à la liste des invités
Paramètres	Nom, prénom, adresse email, numéro de téléphone de l'invité
Pré-condition	L'utilisateur doit être connecté et se trouver dans la partie Invités de l'application
Postcondition	L'invité est ajouté en base de données
Tâche	L'utilisateur appuie sur le bouton "Ajouter" puis saisit les informations relatives à l'invité dans un formulaire. Il valide celui-ci



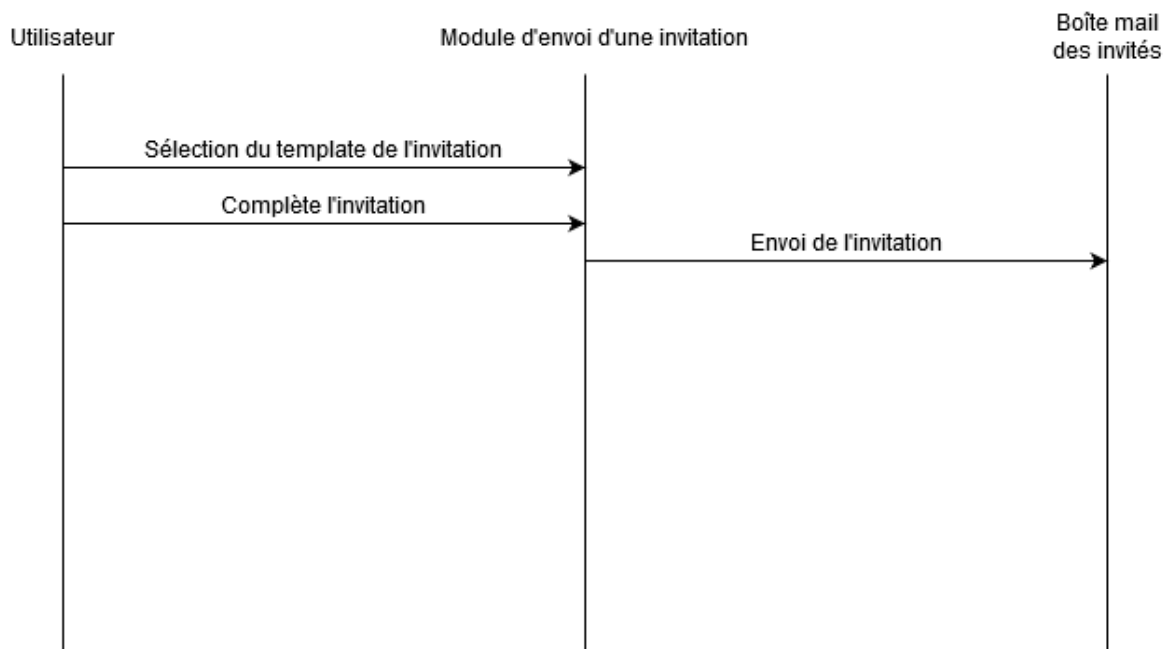
3. Création d'un compte

Traitement	
Rôle	Créer un compte sur l'application
Paramètres	Nom, prénom, pseudo, mail, mot de passe, numéro de téléphone
Pré-condition	L'utilisateur doit se trouver dans la partie Inscription de l'application
Postcondition	L'utilisateur est ajouté en base de données
Tâche	L'utilisateur saisit les informations relatives à son compte dans un formulaire. Il valide celui-ci



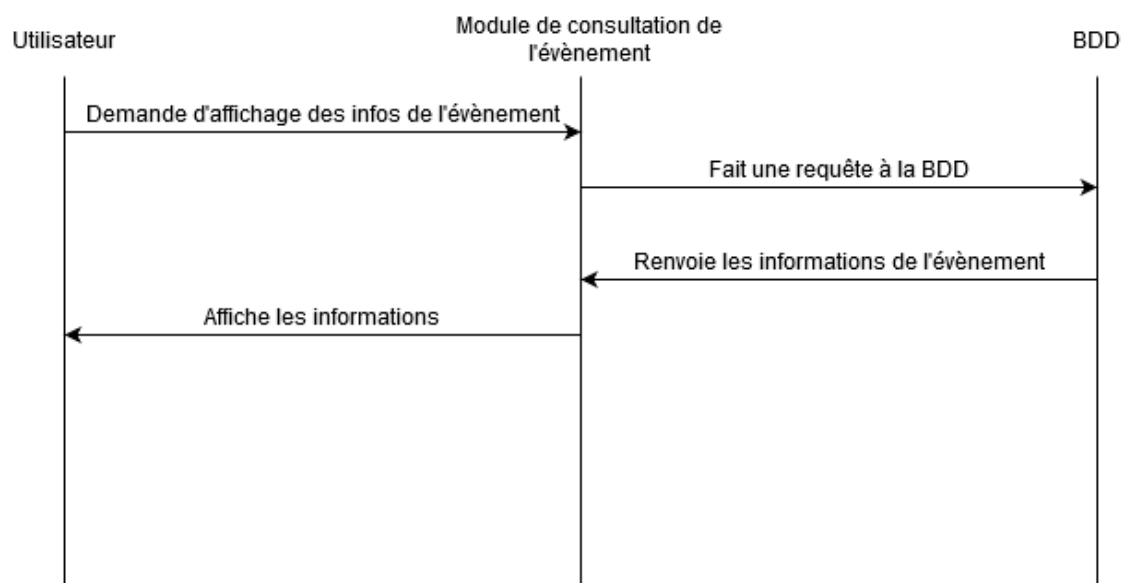
4. Envoi d'une invitation

Traitement	
Rôle	Envoi une invitation à la liste des invités
Paramètres	Message, invité, template de l'invitation, date, lieu
Pré-condition	L'utilisateur doit être connecté et se trouver dans la partie invitation de l'application
Postcondition	L'invitation est envoyé à tous les invités
Tâche	L'utilisateur choisit le template de son invitation puis saisit les informations relatives à cette dernière. Enfin, il l'envoie



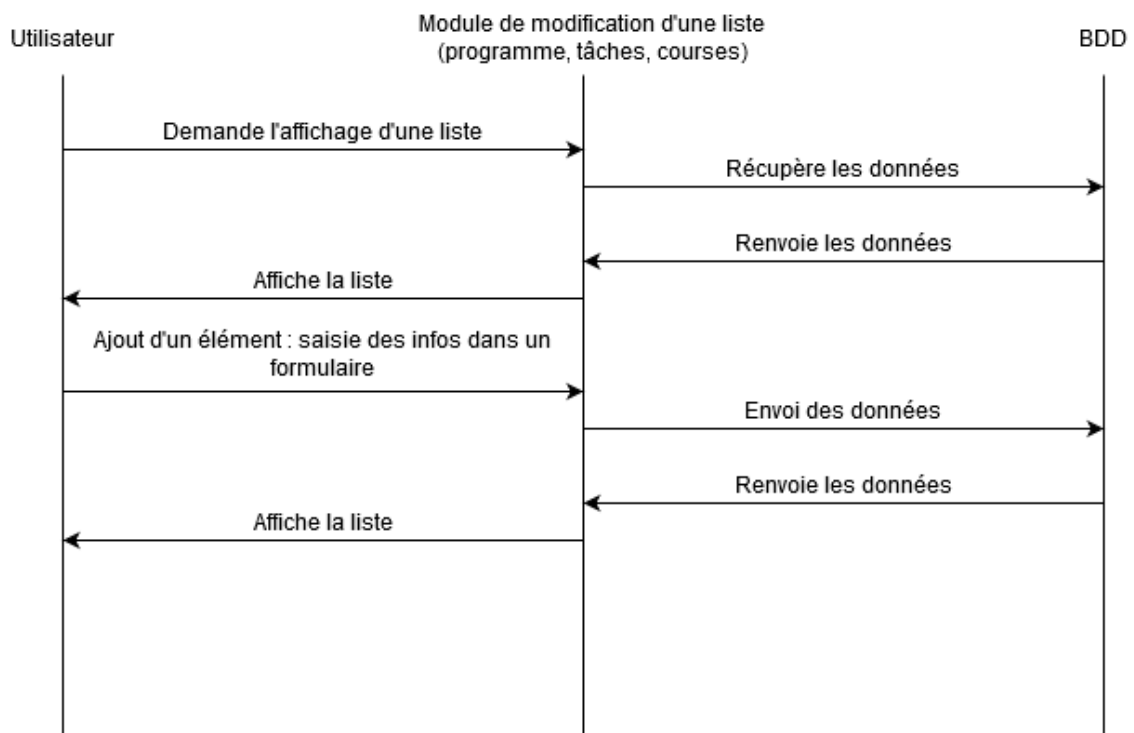
5. Consultation de l'évènement

Traitement	
Rôle	Affiche les informations relatives à l'évènement
Paramètres	Nom, lieu, date et heure de début et de fin, type de l'évènement
Pré-condition	L'utilisateur doit être connecté et se trouver dans la partie "L'évènement" de l'application, l'évènement avoir déjà été créé
Postcondition	
Tâche	Les informations concernant l'évènement comme son nom, date, heure et lieu s'affichent



6. Modification d'une liste

Traitement	
Rôle	Ajoute un élément dans une liste
Paramètres	Intitulé
Pré-condition	L'utilisateur doit être connecté et se trouver dans la Liste souhaitée
Postcondition	L'élément est ajouté en base de données
Tâche	L'utilisateur appuie sur le bouton "Ajouter" puis saisit les informations relatives à l'élément dans un formulaire. Il valide celui-ci



MAQUETTES

Krevent

Créer/Modifier un nouvel évènement

Nom de l'évènement

Lieu de l'évènement

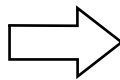
Type

Date de début : 01/03/2020 14:00

Date de fin : 03/03/2020 19:00

Valider

Page de création de l'évènement,
s'affiche suite à la connexion de
l'utilisateur



Krevent

Gérer des évènements

Saisir un budget pour un évènement

Liste des invités

Ajouter une liste de tâches

Ajouter une liste de courses

A propos de l'application

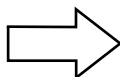
Page d'accueil de l'application une
fois l'évènement créé

Krevent

< Liste des invités

+

Ajouter un invité



Krevent

< Ajouter un invité

Nom de l'invité

Prénom de l'invité

Adresse mail de l'invité

Numéro de téléphone de l'invité

Valider

Page affichant la liste des invités

Page d'ajout d'un invité