

# OPTIMIZER

## Résumé

Notre projet est une application mobile d'organisation personnelle pour aider les étudiants à mieux gérer leurs planning afin qu'ils aient plus de temps pour les activités extra-scolaires et qu'ils puissent partager le contenu qu'ils aiment. Il pourra organiser son temps scolaire comme son temps extra-scolaire, ainsi que se créer un espace de travail cohérent, fluide et adapté.

### Module SES

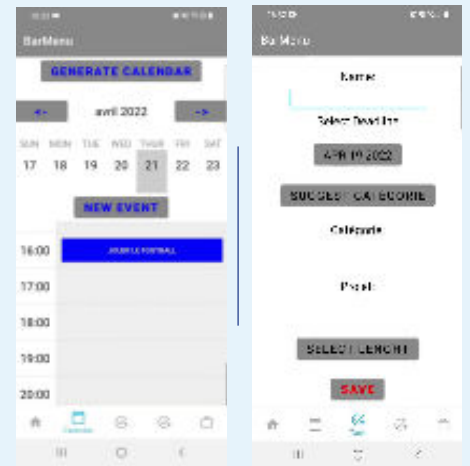
- > Identification des besoins : enquêtes auprès des étudiants
- > Choix des fonctionnalités en fonction du besoin préalablement défini
- > Imagination du squelette de l'application basée sur les retours d'ergonomie des étudiants



### PACT 6.1

BENABBOU Ayoub  
DE MULDER Caroline  
DI COSTANZO Sacha  
DUFFO Yann  
LACROIX Louis  
LECHEVIN Hadrien

### Module ANDROID



- > Création de A à Z d'un emploi du temps mensuel et hebdomadaire
- > Utilisation du langage JAVA pour l'intégration du code de l'algorithme qui génère un emploi du temps.
- > Relier la base de données distante avec l'interface graphique (ajout et suppression de données)
- > Mise en place d'une barre de menu pour une interface fluide et connexion entre les pages
- > Connexion avec le code python

### Module BDD

- > Serveur distant pour le stockage des données
- > Installation d'un serveur Mysql sur une VM avec un serveur Ubuntu
- > Imagination et implémentation de la structure de données dans la base
- > Développement d'une API en PHP pour les communications entre Android et le serveur de BDD



### Module Traitement de texte

- > Associer une catégorie à un mot qui représente une tâche ou un projet
- > Utilisation d'une base de données Wordnet
- > Implémentation de l'algorithme en python et création d'une fonction test



### Module Coeur d'application

#### Modélisation du problème :

- > Système de tâches à faire, que l'on répartit en événements dans le calendrier. Système de repos et de tâches répétitives.
- > Les Time-maps : L'utilisateur choisit hiérarchiquement des créneaux favoris pour ses catégories de tâches favorites.
- > Fonctions de coût d'un calendrier (confort, intensité, écart à la deadline, appartenance aux Time-maps, diversité, ...)
- > Fonctions d'acceptation d'un calendrier qui permettent d'exclure les calendriers inutiles ou impossibles.

#### Résolution du problème :

- > Algorithme heuristique qui répartit les événements de manière uniforme et avec de la diversité entre les tâches.
- > Algorithme génétique qui sélectionne en partant d'un calendrier choisi heuristiquement un meilleur calendrier, avec une reproduction des calendriers qui favorise l'augmentation de la fonction de coût



### Module Test et Intégration

- > Assurer la coordination des différents modules
- > Créer une planification des tâches à réaliser
- > S'assurer de l'intégration de chaque module à l'application finale
- > Tester les fonctionnalités

