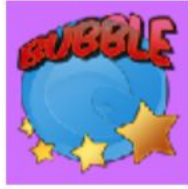


## Rendu 1

### GROUPE B :

Aymé Jacquet  
Guillaume Engel  
Antonio Indrianjafy  
Matthieu Lalbat

## Bubbles :



### Fonctionnalités :

- Système de scoreboard, possibilité du coup de voir ses anciens scores.
- Une requête pour les 10 meilleurs scores
- Une requête pour son meilleur score
- Formulaire de connexion
- Formulaire de création de compte

3 données de login : pseudo, identifiant et mdp

interroge une BDD pour vérifier si l'email est présent dans la BDD

→ Si oui vérifie le mdp

--> Si oui renvoie à l'app OK

→ Envoie le meilleur score de l'utilisateur dans la BDD

- Jeu:
  - Tant que l'utilisateur ne clique pas sur l'écran pour lancer le niveau, le niveau ne se lance pas
  - Créer une variable score initiée à 0
  - Avant que le niveau se lance, le score et le numéro du niveau sont affichés
  - 
  - Création d'un objet Niveau avec les propriétés: Numéro
- Création d'un tableau map dans l'objet niveau avec x nombre entrées (Chaque entrée correspond à une bulle)
  - Dans chaque entrées un objet bulle ou pas
  - Si TRUE alors on place une bulle sur la carte, si FALSE alors on laisse la case vide.
- Créer un objet Bulle: avec les propriétés :ID, Vitesse,Couleur, épaisseur de départ, épaisseur max
  - L'épaisseur de la petite bulle grandit à la vitesse définie
  - Quand l'épaisseur de la petite bulle dépasse l'épaisseur de la grande bulle, la grande bulle est détruite et on perd un certain nombre de points (variable globale au jeu).
  - Si l'utilisateur touche la bulle avant qu'elle soit détruite, la taille de la petite bulle au moment du clic est ajouté au score. Détruire la bulle.
- Le niveau se lance quand l'utilisateur appuie sur le texte
- Une fois que toute les bulles sont détruites,
  - Passer au prochain niveau

- Afficher le score actuel
- Un fois le dernier niveau fini
  - Afficher le score
  - Afficher le meilleurs score
  - Comparer le score au meilleur score de l'utilisateur sur le niveau,
    - Si le score est supérieur, uploader sur la base de donnée et afficher "Bravo new high score"
      - Côté serveur : Update le scoreboard
    - Sinon afficher le meilleur score de l'utilisateur.
    - Afficher les 10 premiers du classement global
- API: Classement entre tous les différents utilisateurs/ Back comprenant des variables email, utilisateur et mot de passe.

Display : "vous êtes connecté, essayez de battre votre meilleur score : X"

### **Solution :**

**Canvas / Phaser JS:** Permet la détection de hitbox (pour les cercles des bulle) et le multitouch. Le canvas permet de charger un niveau rapidement et facilement. L'utilisation de Javascript permet de mettre à jour les scores de façon dynamique et de gérer les informations du serveur plus facilement.

Le fait d'utiliser des technologies web permet de gérer qu'un seul code.

## Memory :



### Côté client :

Détecter de la taille de l'écran

### Côté serveur :

3 données de login : pseudo, identifiant et mdp

interroge une BDD pour vérifier si l'email est présent dans la BDD

→ Si oui vérifie le mdp

→ Si oui renvoie à l'app OK

→ Envoie le meilleur score de l'utilisateur dans la BDD

Display : "vous êtes connecté, essayez de battre votre meilleur score : X"

### Jeu :

- Création d'un tableau niveau dans l'objet niveau avec x nombre entrées (Chaque objet correspond à une carte)
  - Dans chaque entrées un carte
    - Créer des objets carte: avec les propriétés :ID, Couleur,
      - Seul 2 objet peuvent avoir la même couleur

Afficher une grille de cartes.

A l'appui sur une carte :

→ Retourner la carte avec une animation

→ Afficher la carte dans sa couleur secrète, écrire le code hexa de la carte au centre.

→ à l'appui sur une 2ème carte :

→ retourne la 2ème carte

→ affiche sa couleur secrète et son code hexa

→ Si les deux cartes retournées ont la même couleur secrète :

→ gain de points, les cartes restent face visible

→ Variable score

→ Sinon :

→ Si 3ème clic

→ Pas de gain de points

→ Les cartes retournent face caché

Api2: Score.php envoie l'email et le score.

Si score > dernier score : écrase le score

Renvoie la liste des 10 meilleurs scores

**Solution :**

**Corona labs** car spécialisé dans les jeux et développé en OpenGL, ce qui propose une compatibilité avec le web, tout en utilisant les capacités graphiques au maximum. Corona donne accès à une librairie native/ APIS ainsi qu'à d'autres fonctionnalités. C'est un langage très simple et puissant, sachant que le jeu ne nécessite pas énormément de fonctionnalités, c'est une solution appropriée pour cette application(pas de login, scoreboard). Il possède une librairie qui permet de facilement bouger, retourner les objets présents dans le jeu.

## **Fast Tip :**



Côté client :

Le pourcentage du pourboire dans les settings

Pas besoin de se connecter à une bdd, tout ce passe sur l'appareil de l'utilisateur.

Application :

Page Pourboire

Zone de texte, quand on appuie dessus, un clavier apparaît pour pouvoir y rentrer un montant.

Juste en dessous, il y a un bouton "Calcul Pourboire". Lorsque l'on clique dessus, cela calcule le pourboire en fonction du pourcentage entré dans la page "Pourcentage".

Cela affiche également la somme du montant initial et du pourcentage pour avoir un aperçu du montant total.

Page Pourcentage

Dans cette page, une simple zone de texte dans laquelle nous pouvons entrer un pourcentage, et un bouton save pour sauvegarder la donnée dans une variable.

Nous pouvons imaginer un session\_start avec une variable appelée \$\_SESSION['pourcentage'] mis de base à 10. Quand on le change et qu'on sauvegarde, la variable est changée.

**Solution :**

**Ionic**

Etant donné la simplicité de cette application et le fait qu'il n'y est pas de fonctionnalités particulières(Caméra, Fichier, Géolocalisation) ou propres à un système d'exploitation,, la meilleure solution selon nous serait de la développer sur une technologie permettant de déployer sur toutes les plateformes facilement.