

# COMPTE RENDU DE PROJET

## Pop It Game

Application mobile de jeu développée avec Flutter

<b>Technologie</b>	Flutter (Dart)
<b>Type de projet</b>	Application mobile – Jeu interactif
<b>Principe du jeu</b>	Éclater des bulles affichées à l'écran
<b>Difficulté</b>	Progressive – s'adapte au score du joueur
<b>Fonctionnalités clés</b>	Score, vies, game over, historique des scores

## 1. Présentation du projet

Dans le cadre de mon apprentissage du développement mobile avec Flutter, j'ai réalisé une application de jeu inspirée du principe du "Pop It". L'objectif est d'éclater un maximum de bulles affichées à l'écran tout en faisant face à une difficulté qui augmente progressivement en fonction du score. Le jeu intègre un système de vies, un compteur de score, un écran de fin de partie et un historique des scores.

## 2. Fonctionnalités développées

### 2.1 Interface utilisateur – Création des boutons

L'interface repose sur le widget `ElevatedButton` de Flutter, qui permet de créer des boutons interactifs représentant les bulles. Trois propriétés essentielles ont été utilisées :

- `onPressed` : définit l'action exécutée lors du clic sur la bulle (interaction de jeu)
- `style` : personnalise l'apparence du bouton (couleur, forme, marges)
- `child` : contient le contenu affiché (texte ou icône, avec police et couleur configurables)

Ce système a permis de créer une interface claire, interactive et adaptée à l'utilisateur mobile.

### 2.2 Gestion du temps et progression de la difficulté

La difficulté du jeu augmente automatiquement au fil de la partie grâce à la classe `Duration` de Flutter. Une fonction `getCurrentDuration()` a été développée pour adapter dynamiquement la vitesse du jeu en fonction du score :

- Le jeu démarre avec une durée initiale fixe
- Tous les 10 points (10 bulles éclatées), la durée est réduite de 0,05 seconde
- Le calcul suit cette logique : paliers atteints = score / 10, puis réduction = paliers x 0,05 s
- Une durée minimale est définie pour éviter que le jeu devienne injouable
- La fonction retourne la nouvelle durée ajustée à chaque palier

Ce mécanisme rend le jeu progressivement plus rapide et donc plus stimulant pour le joueur.

## 2.3 Gestion des interactions et comptage des bulles

**Chaque clic sur une bulle déclenche une série d'actions :**

- Détection de l'action utilisateur via l'événement `onPressed`
- Incrémentation du compteur de bulles éclatées
- Mise à jour de l'état du jeu en temps réel via `setState()`

Ce mécanisme repose sur la gestion des événements et la mise à jour dynamique de l'interface, deux concepts fondamentaux du développement Flutter.

## 2.4 Système de score et de vies

**Un système de score complet a été mis en place :**

- Le score augmente à chaque bulle éclatée avec succès
- Le score est directement utilisé pour calculer et adapter la difficulté
- Un système de vies (affiché à l'écran sous la forme "Vies : X") pénalise les bulles manquées
- Lorsque toutes les vies sont épuisées, un écran « GAME OVER » s'affiche avec les options Rejouer et Quitter

## 2.5 Historique des scores

**Un écran dédié « Mes Scores » permet de consulter l'historique des parties jouées. Chaque score est enregistré et affiché avec une étoile de notation, offrant ainsi un aspect compétitif et une motivation supplémentaire pour rejouer.**

# 3. Écrans de l'application

Écran	Description
Accueil	Titre du jeu (POP IT GAME) avec les boutons Jouer et Voir mes Scores
Jeu	Grille de bulles bleues à éclater, score et vies affichés en haut de l'écran
Mes Scores	Historique des parties avec affichage du score obtenu
Game Over	Écran de fin avec le score final et les options Rejouer / Quitter

# 4. Compétences mobilisées

Compétence	Mise en oeuvre dans le projet
Framework Flutter	Utilisation de widgets (ElevatedButton, StatefulWidget, etc.)
Gestion des événements	Capture des interactions utilisateur via onPressed
Gestion du temps	Classe Duration, fonction getCurrentDuration()
Interfaces dynamiques	Mise à jour en temps réel avec setState()
Logique de programmation	Calculs de paliers, conditions, fonctions utilitaires
Système de progression	Difficulté croissante basée sur le score du joueur

## 5. Conclusion

**Ce projet m'a permis de concevoir et de développer une application mobile ludique et fonctionnelle avec Flutter. À travers la réalisation du Pop It Game, j'ai appliqué de nombreux concepts du développement mobile : gestion des widgets, des événements, du temps et de l'état de l'application.**

*Le jeu est entièrement jouable, avec une difficulté progressive, un système de vies, un score en temps réel et un historique des parties. Ce projet constitue une expérience concrète et enrichissante dans mon parcours en développement mobile, consolidant ma maîtrise du framework Flutter et de la logique applicative.*

## 6. Illustration



