|  |
| --- |
| Projet JoQuiz |

Table des matières

[1 Analyse préliminaire 3](#_Toc499021832)

[1.1 Introduction 3](#_Toc499021833)

[1.2 Objectifs 3](#_Toc499021834)

[1.3 Planification initiale 4](#_Toc499021835)

[2 Analyse / Conception 4](#_Toc499021836)

[2.1 Concept 4](#_Toc499021837)

[2.2 Stratégie de test 4](#_Toc499021838)

[2.3 Risques techniques 4](#_Toc499021839)

[2.4 Planification 4](#_Toc499021840)

[2.5 Dossier de conception 5](#_Toc499021841)

[3 Réalisation 5](#_Toc499021842)

[3.1 Dossier de réalisation 5](#_Toc499021843)

[3.2 Description des tests effectués 6](#_Toc499021844)

[3.3 Erreurs restantes 6](#_Toc499021845)

[3.4 Liste des documents fournis 6](#_Toc499021846)

[4 Conclusions 6](#_Toc499021847)

[5 Annexes 7](#_Toc499021848)

[5.1 Résumé du rapport du TPI / version succincte de la documentation 7](#_Toc499021849)

[5.2 Sources – Bibliographie 7](#_Toc499021850)

[5.3 Journal de travail 7](#_Toc499021851)

[5.4 Manuel d'Installation 7](#_Toc499021852)

[5.5 Manuel d'Utilisation 7](#_Toc499021853)

[5.6 Archives du projet 7](#_Toc499021854)

*NOTE L’INTENTION DES UTILISATEURS DE CE CANEVAS:  
Toutes les parties en italiques sont là pour aider à comprendre ce qu’il faut mettre dans cette partie du document. Elles n’ont donc aucune raison d’être dans le document final.*

*De plus, en fonction du type de projet, il est tout à fait possible que certains chapitres ou paragraphes n’aient aucun sens. Dans ce cas il est recommandé de les retirer du document pour éviter de l’alourdir inutilement.*

# Analyse préliminaire

## Introduction

Réalisé dans le cadre de mon Pre-tpi, ce projet consiste en une application mobile android pour réaliser des quiz. Ce programme codé en Kotlin, langage nouveau pour moi, me permettra de m’ouvrir a plus d’options pour différents projets à venir.

## Objectifs

* L’application doit être codée en Kotlin
* L’utilisateur doit pouvoir choisir entre plusieurs thèmes donnés
* Une fois le quizz choisi il peut choisir entre plusieurs réponses sous forme d’un QCM
* A la fin du quizz l’utilisateur doit pouvoir voir son meilleur résultat et la date de passation
* Lors de l’ouverture de l’application, les quizz non effectués doivent être affichés en premier
* Le temps de réponse doit également être enregistré
* L’utilisateur doit pouvoir refaire le quizz pour voir les bonnes réponses
* Des questionnaires devraient pouvoir être affichés en ajoutant un fichier de donnée à l’application.

## Planification initiale

Les deux premières semaines seront consacrées à l’analyse du projet

Les 4 prochaines à l’implémentation

Et les deux dernières au tests et documentation

20% d’analyse

50% d’implémentation

10% de Tests

20% de documentation

# Analyse / Conception

## Concept

Voir maquette en annexe

## Stratégie de test

L’application sera testée sur 3 appareils par moi-même et sur 3 autres par :  
  
Joshua Surico

Edward Stewart

Kendra Crystalle Gueissaz

## Risques techniques

Ne maitrisant pas le langage Kotlin pour le moment le plus gros risque est un manque de compétence.  
Je travaille activement sur mon temps libre pour apprendre le langage de mon côté pour palier a ce problème.

## Planification

Semaine 1 **1er février au 2 février**

Planification initiale

Semaine 2 **12 février au 16 février**

Planification initiale

Analyse du projet

Semaine 3 **19 février au 23 février**

Création de l’architecture pour le codeCréation de la base de l’application

Ajout de plusieurs scènes

Semaine 4 **26 février au 1 mars**

Création du system de Quiz en lui-même

Ajout d’un score de fin

Lecture des Quiz par fichier XML

Ajout de Quiz dans l’application

Semaine 5 **4 mars au 8 mars**

Affichage des différents Quiz par score

Enregistrement des scores

Try des Quiz par score max

Semaine 6 **11 mars au 15 mars**

Ajout fonction pour refaire le quiz en affichant les bonnes réponses

Ajout de vision de l’historique des scores par quiz

Semaine 7 **18 mars au 23 mars**

Test du projet

Finalisation du design

Semaine 8 **25 mars au 29 mars**

Correction de bug connu

Documentation du projet

## Dossier de conception

*Fournir tous les document de conception:*

* *le choix du matériel HW*
* *le choix des systèmes d'exploitation pour la réalisation et l'utilisation*
* *le choix des outils logiciels pour la réalisation et l'utilisation*
* *site web: réaliser les maquettes avec un logiciel, décrire toutes les animations sur papier, définir les mots-clés, choisir une formule d'hébergement, définir la méthode de mise à jour, …*
* *bases de données: décrire le modèle relationnel, le contenu détaillé des tables (caractéristiques de chaque champs) et les requêtes.*
* *programmation et scripts: organigramme, architecture du programme, découpage modulaire, entrées-sorties des modules, pseudo-code / structogramme…*

### Choix de l’IDE

L’application sera intégralement codée depuis l’IDE Android studio Code.

C’est l’IDE officiel de google pour le développement Android. Et prend en charge nativement Kotlin ce qui me donne accès a beaucoup de fonctionnalité intéressante pour notamment le débogage.

Android studio possède également un émulateur android inclus ce qui me permettra de tester JoQuiz sur une multitude d’OS et format d’écran.

### Base de données

JoQuiz utilisera comme base de données plusieurs fichier XML. Un pour faire l’historique des scores accompli sur les différents quiz et récupèrera dans un dossiers plusieurs fichiers XML créé par l’utilisateur qui contiendront les quiz et question en eux-mêmes

### Architecture du projet

*Coming soon*

### Diagramme de flux du point de vue utilisateur

Voir annexe

# Réalisation

## Dossier de réalisation

*Décrire la réalisation "physique" de votre projet*

* *les répertoires où le logiciel est installé*
* *la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)*
* *les versions des systèmes d'exploitation et des outils logiciels*
* *la description exacte du matériel*
* *le numéro de version de votre produit !*
* *programmation et scripts: librairies externes, dictionnaire des données, reconstruction du logiciel - cible à partir des sources.*

*NOTE : Evitez d’inclure les* *listings des sources, à moins que vous ne désiriez en expliquer une partie vous paraissant importante. Dans ce cas n’incluez que cette partie…*

### Version d’Android

Pour créer mon projet j’ai choisi comme SDK minimum l’api 21 qui corresponds à Android 5.0 (Lollipop version).

Cela a pour but que l’application puisse fonctionner sur le maximum de téléphone android (99%).

### Création des scènes

J’ai commencé par créer toutes les scènes de mon projet afin de faciliter le développement de la suite de mon projet.  
Séparé en 4 fichiers ces « Activité » sont codé en XML grâce au mode « Design » d’Android studio et stocké dans le dossier « layout ».

* Activity\_main.xml
* Historique.xml
* Quiz.xml
* Résultat.xml

Voir images annexes

### Navigation entre différentes scènes

Afin de pouvoir naviguer entre différentes scènes a l’aide de bouton au cours du projet j’ai créé la classe NavigationHandler plutôt que de placer dans chaque classe ou on trouve de la navigation la même fonction.

Elle est composée de quatre fonctions :

* **setupButtonNavigation** : Attribue une action de navigation à un bouton spécifique, permettant le passage à une nouvelle activité définie. Cette méthode associe dynamiquement un bouton à une page cible, initiée lors de l'activation du bouton.
* **goToFirstQuiz** : Navigue vers la scène la première scène de l’activité “Question”, tout en communiquant a la classe correspondante l’id du quiz cliqué.
* **quizHandler** : Fonctionne comme goToFirstQuiz, avec l'ajout d'une variable permettant de transmettre à quel question l’utilisateur se trouve et une autre passant l'état des réponses de l'utilisateur sous forme d'un tableau de booléens (BooleanArray) afin de former à la fin les résultats du quiz.
* **goToResult** : Navigue vers la scène de résultat en communiquant notre tableau de booléens et l’id du quiz afin d’afficher les informations requises.

### Chargement des quiz depuis un fichier xml

Afin de gérer l’ajout de plusieurs quiz, j’ai concu la classe la classe QuizRepository pour charger les données des quiz à partir d'un fichier XML. Cela centralise le processus de chargement et de stockage des quiz, évitant ainsi la redondance du code et facilitant la maintenance.

Voici les fonctions principales du code, réutilisé dans le reste du projet :

* **loadQuizzesFromXML** : Lit le fichier XML, extrait les quiz, leurs questions et réponses, puis les stocke.
* **parseQuestion** : Analyse chaque question du fichier XML pour récupérer le texte, la bonne réponse et les mauvaises réponses.
* **getAllQuizzes** : Donne accès à la liste des quiz chargés pour être utilisés dans l'application.

Cette classe rend l'accès aux quiz facile et organisé pour une meilleure expérience utilisateur.

## Description des tests effectués

Afin d’assurer que l’application fonctionne correctement et qu’aucun crash ne survienne j’ai fait tester mon application a plusieurs personnes de mon entourage sur divers téléphone Android j’ai partagé mon application.  
Ayant au départ choisi de faire tester mon application a seulement trois personnes, j’ai finalement opté à faire tester JoQuiz au maximum de personnes possible pour avoir un résultat beaucoup plus précis.

**Les bugs suivant ont été relevé :**

bug récursivité menu principal recycleur ; voir annexe

bug récursivité menu historique recycler ; voir annexe

Problème d’affichage titre quand on lance un quiz ; voir annexe

Ces trois problèmes ont pu donc être résolu ; voir annexe

Propriétaire et Appareil sur lequel l’app a été testé :  
  
Samsung S23 : Fernando Chirino

Samsung S23 Ultra : Levy Stoller

*Pour chaque partie testée de votre projet, il faut décrire:*

* *les conditions exactes de chaque test*
* *les preuves de test (papier ou fichier)*
* *tests sans preuve: fournir au moins une description*

## Erreurs restantes

Bug en mode paysage :  
  
L’application n’a pas été adaptée au mode paysage du téléphone. Voir annexe

Cela pour conséquence que si l’utilisateur tourne son téléphone pour une raison ou une autre et qu’il a le mode paysage activé l’application ne pourras pas s’afficher correctement.  
J’envisage de faire une version de mes activités spécialement pour le mode paysage, Il est également possible de simplement désactiver cette option afin d’éviter toute confusion chez l’utilisateur.

*S'il reste encore des erreurs:*

* *Description détaillée*
* *Conséquences sur l'utilisation du produit*
* *Actions envisagées ou possibles*

## Liste des documents fournis

*Lister les documents fournis au client avec votre produit, en indiquant les numéros de versions*

* *le rapport de projet*
* *le manuel d'Installation (en annexe)*
* *le manuel d'Utilisation avec des exemples graphiques (en annexe)*
* *Autres…*

# Conclusions

*Développez en tous cas les points suivants:*

* *Objectifs atteints / non-atteints*
* *Points positifs / négatifs*
* *Difficultés particulières*
* *Suites possibles pour le projet (évolutions & améliorations)*

# Annexes

## Résumé du rapport du TPI / version succincte de la documentation

## Sources – Bibliographie

**Android Tutorial (Kotlin) - 30 - SQLite Database Creation and Insertion**

De : CodeAndroid

<https://youtu.be/OxHNcCXnxnE?si=pR-mbGz_Gx66D086>

**How to develop app to parse XML data - Android Kotlin**

De : Dr. Parag Shukia

<https://youtu.be/bPtrRCkPPFo?si=t3ILcLP9SFtR2rpQ>

**RecyclerView With Item Click Listener in Kotlin : ( Android Tutorial 2022 )**

De : CodingSTUFF

<https://youtu.be/WqrpcWXBz14?si=Yct-5GrvYRiwS221>

**Creating a RecyclerView that handles Click Events in Android Studio (Kotlin 2020)**

De : Indently

<https://youtu.be/ai9rSGcDhyQ?si=MupczYM0hhrfGUdD>

**Kotlin Course - Tutorial for Beginners**

De : freeCodeCamp.org

<https://youtu.be/F9UC9DY-vIU?si=9Ud6biwNBrvoh6N4>

<https://www.practicalcoding.net/blog/categories/kotlin>

## Journal de travail

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |
|  |  |  |  |

## 

## Manuel d'Installation

## Manuel d'Utilisation

## Archives du projet

*Media, … dans une fourre en plastique*