Table des matières

[Objet du document 2](#_Toc5955254)

[Présentation du projet 2](#_Toc5955255)

[Définition du Besoin 2](#_Toc5955256)

[Analyse du besoin 2](#_Toc5955257)

[Les contraintes liées au projet 2](#_Toc5955258)

[1. Ergonomie 2](#_Toc5955259)

[2. Environnement 2](#_Toc5955260)

[3. Architecture 3](#_Toc5955261)

[4. Durée de vie 3](#_Toc5955262)

[Objectif de la mission 3](#_Toc5955263)

[Planning, Livrable et Groupe de travail 4](#_Toc5955264)

[Outils et Environnement utilisés 4](#_Toc5955265)

[Analyse et conception 5](#_Toc5955266)

[Diagramme de cas d’utilisation 5](#_Toc5955267)

[UC1(Répondre au calcul) 5](#_Toc5955268)

[UC2(Consulter l’aide) 5](#_Toc5955269)

[Interface Homme-Machine (IHM) 5](#_Toc5955270)

[Diagramme des séquences EDITCALC v1 7](#_Toc5955271)

[Diagramme d’état (semi-détaillé) EDITCALC v2 8](#_Toc5955272)

[Réalisation et tests 8](#_Toc5955273)

[Photo du code 8](#_Toc5955274)

[Tests 8](#_Toc5955275)

[Annexes 8](#_Toc5955276)

# Objet du document

## Présentation du projet

C’est un projet se déroulera en 2 phases.

Le but du projet est de développer une application ludique à destination des enfants de classes de « CP1 » afin qu’ils apprennent à additionner et soustraire jusque 10.

L’application devra être déployée auprès de 1200 écoles, dans les classes de CP1 et sur 15 ordinateurs (de chaque école) (total (en considérant une seule classe de 1 classe CP1 par école) : 180 000 déploiements.

## Définition du Besoin

L’objet est de développer une application qui doit répondre à ce besoin « une calculette bi-opération (x,y) à restitution calculo-imagée »

Si le langage utilisé pour développer le programme reste à l’appréciation du développeur, Le développement de l’application est soumis à plusieurs contraintes du fait de la destination de cette application.

## Analyse du besoin

Dans le cadre de ses activités, l’éditeur « EditCALC » propose des solutions éducatives pour les écoles.

L’une de ces solutions éducatives possède 2 fonctions.

Les clients cibles sont des écoles et les classes préparatoires première année (CP1) où on apprend à compter jusque 10.

## Les contraintes liées au projet

Si le langage utilisé pour développer le programme reste à l’appréciation du développeur, Le développement de l’application est soumis à plusieurs contraintes du fait de la destination de cette application.

### Ergonomie

L’application étant destiné à un public très jeune (enfant en classe de CP1), il faut que l’application soit très simple à être utilisée, mais, il faut aussi qu’elle soit ludique, afin de donner l’envie à l’enfant d’utiliser l’application.

* Le calcul affiché devra être imagé (par exemple une animation en cas de réussite et une différente en cas d’échec)
* Le calcul affiché pourra proposer des figurines statiques et/ou animées.
* Une aide en ligne doit pouvoir être disponible (un bouton « ? »)

### Environnement

L'application devra pouvoir être utilisé sur PC (windows), à partir de la version Windows 7, sur Tablette Android et Windows mobile.

### Architecture

Le service doit permettre :

* la saisie
* la visualisation à l’écran
* la configuration des jeux d’image

### Durée de vie

L’application devra être maintenue durant 3 années après la date de première livraison.

## Objectif de la mission

La mission sera décomposée en 2 phases.

* La première phase ou V1 devra répondre aux exigences suivantes :

L’application doit afficher, au démarrage, une publicité aléatoire. Après 3 secondes d’affichage, l’application « calculatrice » sera disponible pour l’enfant, et, il pourra saisir le résultat qu’il pense être le bon. Le résultat restera affiché 5 secondes et l’application se fermera automatiquement.

Afin de répondre à un autre calcul, l’enfant devra relancer l’application.

* La seconde phase ou V2 devra être une évolution de la V1 et devra répondre aux exigences suivantes :

L’application doit afficher, au démarrage, une publicité aléatoire. Après 10 secondes, l’application « calculatrice » sera disponible pour l’enfant, et, il pourra saisir le résultat qu’il pense être le bon.

Ce qui diffère avec la phase 1 (ou V1) :

L’enfant a trois (3) chances pour donner le bon résultat.

Cas où l’enfant donne la bonne réponse dans le nombre d’essai possible :

Message de félicitation.

Sinon, message pour réessayer.

A l’issu, le résultat est donné (affiché).

L’enfant peut choisir de quitter l’application ou de recommencer.

S’il quitte l’application, alors une publicité s’affiche pendant 5 seconde.

Cas où l’enfant donne 3 mauvaises réponses, le bon résultat devra être affiché.

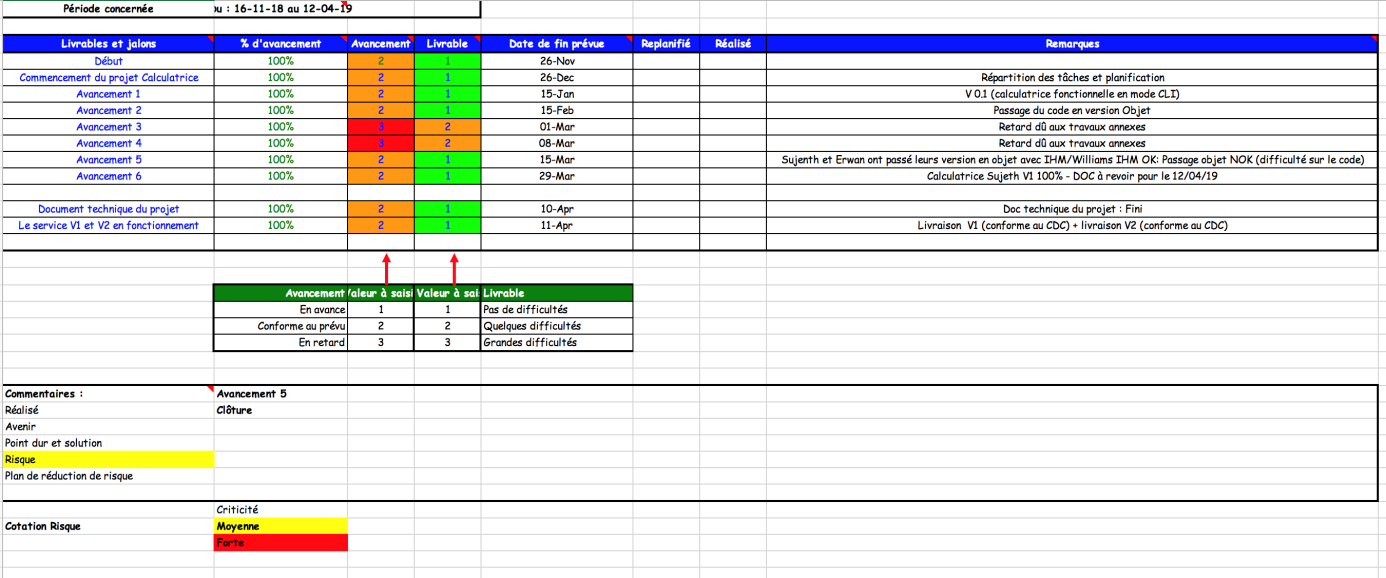
## Planning, Livrable et Groupe de travail

Nous avons travaillé à distance, nous étions un groupe de 3 personnes (Erwan, Sujenth, Williams).

Nous utilisions le moyen de communication : DISCORD

Afin d’échanger sur nos avancées respectives et nous échanger les différents documents nécessaires.

Les risques identifiés et avérés ont surtout été un risque de délai à cause des travaux annexes à ce projet (BTS Blanc, préparation épreuve U22….)



## Outils et Environnement utilisés

L’outil de développement utilisé est Eclipse.

Nous avons choisi de développé l’application dans un environnement Windows, pour une meilleure compatibilité avec les environnements dans lesquels l’application sera déployée.

## Les outils

Après une réunion de groupe, nous avons décidé d’utilisé JAVA et l’IDE Eclipse pour développer notre application, JAVA est un langage strict et permet donc une maintenance plus aisé pour une personne ou groupe qui serait amener à maintenir l’application et qui n’auraient pas développé l’application, de plus si une évolution serai à prévoir JAVA étant un langage objet et qui dit objet dit réutilisabilité donc notre choix naturel s’est porté sur JAVA.

## 5. Contraintes

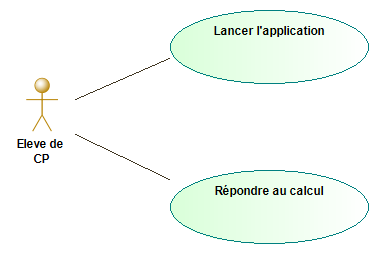
Les contraintes que nous avons pu lister après analyse du cahier des charges sont les suivantes :

* Le calcul à deviner ne doit pas dépasser 10
* Deux opérations possibles seulement(+, -)
* Mettre des dessins pour enfant
* Mettre des pubs aux moments prévus
* Les enfants doivent répondre à l’aide d’une IHM et non avec le clavier

# Analyse et conception

## Diagramme de cas d’utilisation

Nous avons voulu utilisé un cas d’utilisation afin d’être compris de tous.



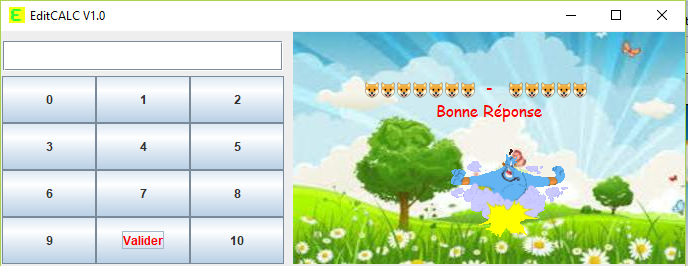
### UC1(Démarrer l’application)

L’élève devra démarrer l’application.

### UC2(Proposer un résultat)

L’élève de CP devra répondre au calcul demandé. Pour ce fait il devra choisir un nombre entre 0 et 10 puis valider. Pour modifier sa réponse, il lui suffit de choisir un autre nombre avant de valider.

## Interface Homme-Machine (IHM)



4a

3

1

4b

2a

2b

Légende :

Partie Calculatrice :

1 🡪 Zone de texte : Affiche le nombre sélectionné

2a 🡪 Boutons (0-10)

2b 🡪 Bouton Valider : Valide la réponse choisie et vérifie si elle est juste

Partie Graphique :

3 🡪 Calcul : Affiche sous forme de chiens pour que l’élève apprenne à compter et à bien vérifier le résultat.

4a 🡪 Affiche « Bonne réponse » ou « Mauvaise réponse » selon le résultat proposé

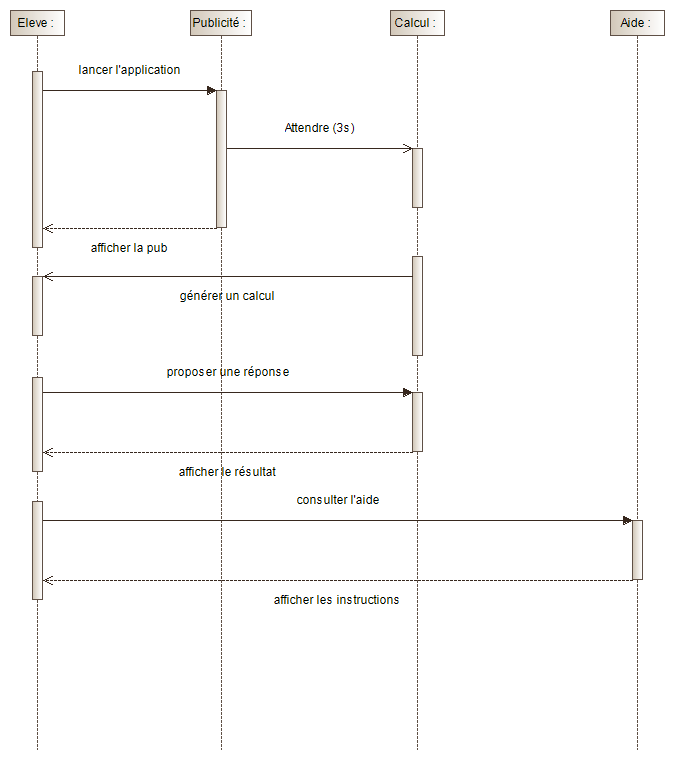
4b 🡪 Gif : Pour rendre l’application plus ludique

Principe :

L’élève devra répondre au calcul affiché dans la partie graphique. Il n’aura qu’un seul essai dans cette version. Il devra choisir une réponse entre 0 et 10 qu’il pourra modifier à volonté en cliquant sur le bouton correspondant au nombre voulu avant de valider sa réponse. L’élève attendra 3 secondes avant de savoir si son résultat est juste ou pas.

## 

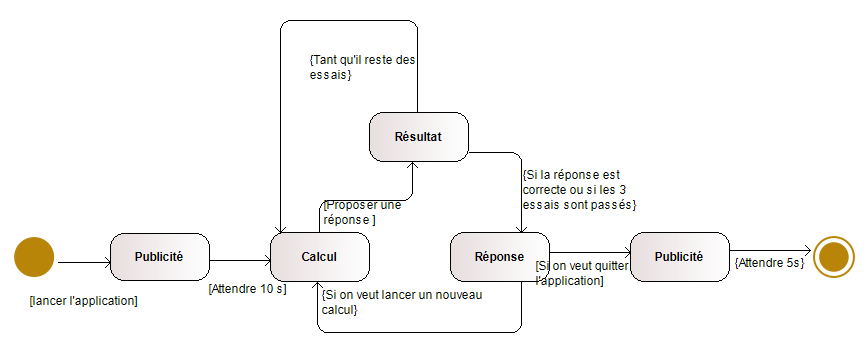
## Diagramme des séquences EDITCALC v1



Nous avons utilisé ce diagramme de séquence afin d’avoir un aperçu clair de l’application.

Une fois l’application lancée par l’élève, il devra visionner une publicité (pour enfants) animée ou non de trois secondes. Ensuite un calcul sera généré. L’élève pourra pendant ce temps, consulter le mode d’emploi pour apprendre à utiliser l’application. Il devra proposer une réponse et verra si son résultat elle est juste ou fausse, sans pour autant connaître la réponse. L’application se fermera juste après.

## Diagramme d’état (semi-détaillé) EDITCALC v2



L’évolution entre la version 1 et version 2 est assez conséquente tout en conservant les fondements du processus de l’application. La publicité d’ouverture dure dix secondes cette fois-ci. L’élève aura trois tentatives pour trouver la bonne réponse. La réponse s’affichera si l’élève aura trouvé la bonne réponse ou si les trois essais sont passés. L’élève pourra choisir entre relancer un nouveau calcul ouquitter l’application. A la fermeture de l’application, une publicité de fermeture s’affichera pendant cinq secondes.

# Réalisation et tests

## Photo du code

## Tests

Les problèmes rencontrés ont été pour beaucoup du aux techniques employées. Par exemple, le système de Timer de l’application afin de gérer le temps d’attente entre les différentes vues de l’IHM.

Un autre a été la coordination entre les ressources humains (différents rythmes de travail...).

Exemple à faire pour V1 et V2

|  |  |
| --- | --- |
| **Problème rencontré/ Anomalie** | **Résolution/Action** |
| Coordination des ressources humaines | Utilisation d’application de communication (discord) |
| La pub devait durer 10s mais c’est l’application qui se mettait en pause. | Correction de la gestion des threads. |
|  |  |

## Annexes

# Déploiement

Nous avons décidé de déployer l’application via un site web simple. L’accès sera sécurisé par un système « login/password ». Après authentification l’utilisateur pour alors télécharger le fichier exécutable.

NB : Par soucis de temps cette partie n’a pas été réalisée.

# VII. Amélioration possible