|  |
| --- |
| Projet I Shoes |



Romain Lenoir – responsable-développement

Sacha Jaccard – Responsable maquettes

Alex Barreira-videira – Scrum master

Table des matières

[1 Analyse préliminaire 3](#_Toc130306947)

[1.1 Introduction 3](#_Toc130306948)

[1.2 Objectifs 4](#_Toc130306949)

[1.3 Planification 5](#_Toc130306950)

[2 Analyse / Conception 5](#_Toc130306951)

[2.1 Concept 5](#_Toc130306952)

[2.2 Stratégie de test 6](#_Toc130306953)

[2.3 Planification 6](#_Toc130306954)

[2.4 Dossier de conception 6](#_Toc130306955)

[3 Réalisation 8](#_Toc130306956)

[3.1 User case 8](#_Toc130306957)

[4 Conclusion 9](#_Toc130306958)

[5 Annexes 9](#_Toc130306959)

[5.1 Sources – Bibliographie 9](#_Toc130306960)

[5.2 Journal de travail 9](#_Toc130306961)

[5.3 Archives du projet 9](#_Toc130306962)

# Analyse préliminaire

## Introduction

I Shoes est un site d’e-commerce spécialisé dans la vente de chaussure. Il est réalisé dans le cadre d’un projet d’e-commerce pour le cours de Projet Web des deuxièmes années du CPNV. Ce site permettra aux utilisateurs d’entretenir un compte avec son propre panier et sa propre empreinte d’achat. Il sera ainsi possible de naviguer sur tout le site afin d’y rencontrer nos derniers articles. Un compte administrateur est également propice où l’utilisateur pourra créer un nouvel article, y rajouter son stock et à en plus de cela accès à un historique d’achat. Une vue ‘utilisateur’ est également mis pour l’administrateur du site.

## Objectifs

L’objectif du projet est d’avoir les options ci-dessous :

* Pouvoir accéder au site.
* Pouvoir créer un nouveau compte utilisateur.
* Pouvoir se connecter aux comptes utilisateurs précédemment créer.
* Avoir un page de descriptif entre chaque articles.
* Concevoir une page administrateur pour l’administration des articles et d’informations sur le site.
* Avoir un historique d’achat des utilisateurs pour l’administrateur lui-même.

## Planification

La planification de ce projet est conçu et planifié avec l’aide du site icescrum, dont l’url se trouve ici :

*https://icescrum.cpnv.ch/p/PROJWEB202/#/project*

# Analyse / Conception

## Concept

Pour le concept du site, on a procédé de la façon suivante :

Les recueils de données se font avec des pages JSON :

* dataUsers.json => Sert de recueil pour les utilisateurs et comptes prochainement créer et administré sur le site.
* dataArticles.json => Sert de recueil pour les différents articles du site. Cette page ne peut qu’être modifié par l’adminitrateur du site.
* DataBasket.json => Sert de recueil pour les différents articles mis sur le panier des utilisateurs.
* DataPurchases => Sert de recueil pour les articles achetés sur le site. Cette page est également utilisé pour la page d’historique d’achat.

Les maquettes du site sont les suivantes :

Les maquettes visionnables en vue ‘ordinateur’ :

Mettre ici les maquettes…

Les maquettes visionnables en vue ‘mobile’ :

Mettre ici les maquettes…

Les maquettes de connexion et d’inscription en vue ‘ordinateur’ :

Mettre ici les maquettes…

Les maquettes de connexion et d’inscription en vue ‘mobile’ :

Mettre ici les maquettes…

## Stratégie de test

Description de la stratégie globale de test :

* Si un utilisateur lambda clique sur un article, l’utilisateur se retrouve sur la page de l’article spécifié.
* Si l’utilisateur met son article demandé sous son panier, il se glisse sous celle-ci.
* Si l’utilisateur envoie son panier, il est ensuite en cours de payement et l’utilisateur se voit redirigé sur une page d’achat.
* Si l’utilisateur n’inscrit pas les bons logges, un message d’erreur s’affiche.
* Si l’utilisateur créer un compte utilisateur qui n’est pas unitaire, un message d’erreur s’affiche.
* Si l’utilisateur essaie d’ajouter son article dans son panier alors qu’il n’est pas connecté par son compte, un message d’erreur s’affiche et l’utilisateur est redirigé sur la page php ‘login’.
* Si l’utilisateur se voit malencontreusement à mettre dans son panier un article qui n’a pas de stock ou qu’il en demande plus que cet article comporte, un message d’erreur s’affiche.
* SI l’administrateur créer un article avec une photo non-négligeable, un message d’erreur s’affiche.
* Si l’administrateur est connecté, il aura accès à la page d’administration du site. Il pourra créer des articles et y rajouter leur stock.

## Planification

Révision de la planification initiale du projet :

* Le projet a commencé le 02.02.2023 et terminera duramt la semaine du 4 avril 2023.
* Alex est devenu Scrum master.
* Romain est responsable développement et a aidé pour la création et finition de la documentation.
* Sacha est responsable des maquettes à concevoir et de la documentation du projet.

## Dossier de conception

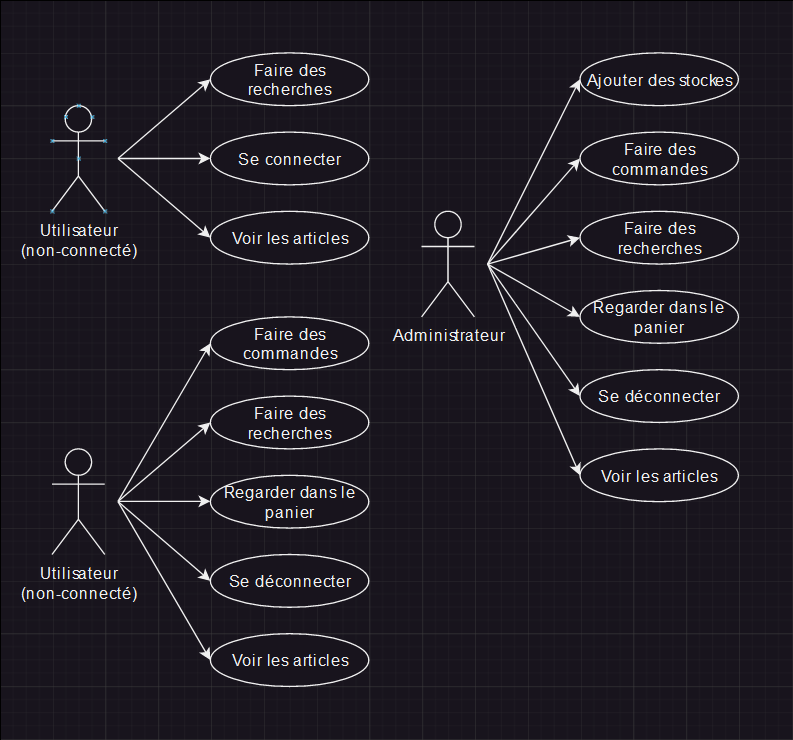
* Pour ce dit projet, le système d’exploitation Windows 10 est utilisé.
* Pour les maquettes du projet, ‘Balsamiq’ est le logiciel utilisé.
* Pour la réalisation de diagramme de flux, le logiciel et site internet ‘Draw.io’ est utilisé
* Pour la réalisation des journaux de travail, Excel est le logiciel utilisé.
* Pour la réalisation et structure du code, le php et l’interpréteur php est utilisé.
* Pour la documentation et notes du projet, Word est le logiciel utilisé.
* Pour l’exportation des différentes maquettes, du Wireframe et des user-cases, le format PDF est utilisé.
* Pour la structure des bases de données, nous avions sobrement voulu que cette structure soit simple et qu’il soit également facile à comprendre.

# Réalisation

Réalisation des modèles de diagrammes de flux :

Mettre le diagramme ici…

## User case



# Conclusion

Pour conclure la réalisation du projet, nous avions remplit tous nos objectifs. Pour ce qui sera des problèmes rencontrés, nous avions eu certains problèmes avec la suppression et diminution en ce qui concerne les divers articles. Je pense, plus précisément, à l’utilisateur qui enlève de lui-même un stock d’un article avant de l’avoir acheté. Également, je pense aux problèmes que nous avions eu en créant un fichier .json pour le panier et un fichier .json pour les articles achetés. Le problème étant de bien administrer les deux en même temps, vu que l’un est aussi dépendant avec l’autre que son prochain => Ex : quand un utilisateur achète l’article en question, il est déjà inscrit sous sa page .json attribuée avant d’être supprimé sur la page de panier. A part cela, nous avions eu de grands espoirs pour ce projet et avions acquis des compétences au niveau de la compréhension du code et de réalisation de projet grâce à ceci.

# Annexes

## Sources – Bibliographie

*Romain Lenoir =>*

[*https://forums.commentcamarche.net/forum/affich-9781202-css-force-retour-a-la-ligne*](https://forums.commentcamarche.net/forum/affich-9781202-css-force-retour-a-la-ligne)

[*https://www.w3schools.com/php/func\_array\_splice.asp*](https://www.w3schools.com/php/func_array_splice.asp)

*Sacha Jaccard =>*

*…*

*Alex Barreira-videira =>*

*…*

## Journal de travail

Pour ce projet, nous avions conçu trois différents jounaux de travail. Ces journaux sont personnels.

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

## 

## Archives du projet

*Archive GitHub du projet :* [*https://github.com/romain-lnr/I\_Shoes*](https://github.com/romain-lnr/I_Shoes)