

— — — S3.01 — — —

Département informatique

Anthony HASCOET

Mateo FLEJOU

Clément MONFORT

Veille technologique

-

Balabox - Outils ladigitale.dev

— — — — — — — — —



**BalaBox**

Collège  
**N**umérique 56

<b>Introduction</b>	<b>2</b>
<b>Techniques</b>	<b>2</b>
Outils	2
Technologies	3
WebSocket	4
Présentation	4
Cas d'utilisation	4
<b>Comparaison Application</b>	<b>4</b>
Digistorm	4
Digiscreen	5
Digiboard	6
Le cas Openboard	6
Le cas Miro	6
<b>Conclusion</b>	<b>7</b>

# Introduction

Le but d'une **veille technologique** ou **veille scientifique technique** est de s'informer sur les techniques les plus récentes et leur mise à disposition commerciale. Notre document va principalement faire l'état de l'art des **applications Web** et des **technologies** utilisées par ses dernières.

Contrairement aux autres sous-projets, nous devons implémenter 8 applications déjà fournies par le client dans la [BalaBox](#). Par conséquent, nous avons peu de choix sur les technologies utilisées car déjà prédéterminé par le travail de [La Digitale](#). De plus, nous couvrirons seulement les 3 applications prioritaires qui ont été définies dans la Proposition de réalisation.

Voici les applications concernées :

1. [Digiboard](#) : permet de créer des tableaux blancs collaboratifs où les élèves et professeurs pourront ajouter des éléments et les modifier.
2. [Digiscreen](#) : permet de créer un fond d'écran interactif.
3. [Digistorm](#) : permet de créer et d'animer des sondages, remue-ménages, questionnaires et des nuages de mots.

# Techniques

## Outils

Pour mener à bien ce projet, nous utilisons [GitLab](#) à la fois comme **gestionnaires de versions** ainsi que comme un **outil d'organisation** en tandem avec [Notion](#). Le premier pour communiquer l'avancée du développement (Les problèmes, leurs résolutions, l'état des fonctionnalités, ...), et le second pour répartir et organiser la charge de travail.

Pour développer l'application, nous utiliserons deux IDE (Environnement De Développement): [Visual Studio Code](#) pour les applications serveurs (ceux de [La Digitale](#)) et [Androïde Studio](#) pour l'application client.

## Technologies

Les applications de [La Digitale](#) sont techniquement terminées car notre mission consiste à un portage de ses applications en l'état dans l'environnement de la [BalaBox](#), donc par faute de pouvoir justifier les technologies choisies, nous allons les expliquer.

Les applications sont presque entièrement basé sur le langage de scripte ([langage de programmation interprété](#)) [JavaScript](#) la colonne vertébrale de toute les applications web moderne :

- Les bases de données sont gérées par [Redis](#), un gestionnaire de [base de données à clé-valeur](#) (Une [Table de hachage](#)) [extensible](#) en [NoSQL](#) ([C ANSI](#)) sous [licence BSD](#).
- Le back-end des applications est basé sur [Express.js](#), un [framework](#) standard pour construire des serveurs en [Node.js](#), une [plateforme logicielle libre](#), soit un environnement d'exécution (Machine virtuelle) en [JavaScript](#).
- Le front-end des applications est basé sur du [CSS](#) et [Nuxt.js](#) un [framework](#) qui rend l'utilisation du [framework](#) [Vue.js](#) plus intuitive, [Vue.js](#) qui est lui même basé sur [Node.js](#).
- L'application de lancement des applications sera elle aussi en [Node.js](#).
- Cependant l'application Cliente sera réalisée sur [Androïde Studio](#) en [Kotlin](#).

## WebSocket

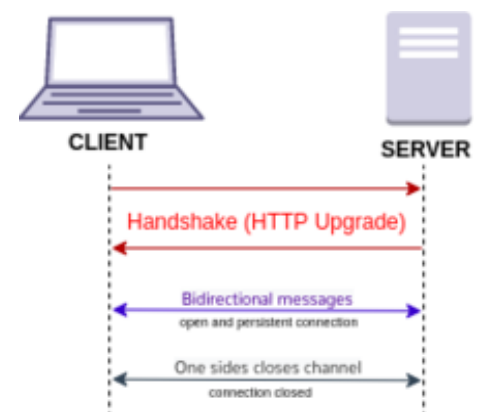
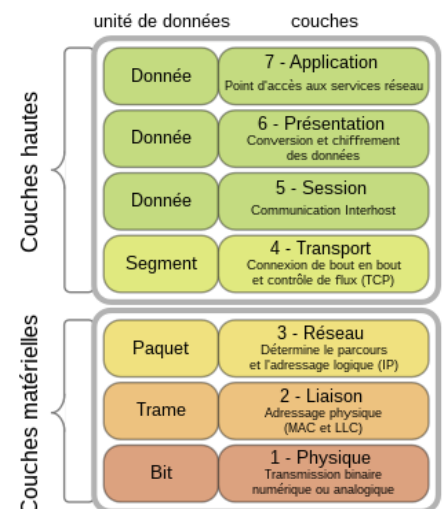
Il s'agit du principal protocole de communication utilisé dans les applications de [La Digitale](#) que nous serons amenés à manipuler lors de leurs portages.

### Présentation

Les WebSockets sont un standard Web, désignant à la fois un protocole réseau de la couche application (7<sup>e</sup> couche du modèle OSI), ainsi qu'une **API** (interface de programmation) du **World Wide Web**, permettant d'ouvrir un canal de communication en **full-duplex (bidirectionnelle)** se basant sur un **socket TCP** pour les **navigateurs** et **serveurs web**.

### Cas d'utilisation

Ils sont principalement implémentés pour leur **faible temps de réponse**, la **notification** au **client** d'un **changement d'état du serveur** ainsi que l'envoi de données en méthode **Push** du serveur vers le client **sans requête**. Ils sont utilisés dans toutes les applications web interactives (messagerie instantanée) et permettent de contourner les nombreux **obstacles réseaux** comme les **pare-feu**.



## Comparaison Application

### Digistorm

L'avantage de [Digistorm](#) par rapport à d'autres applications similaires est qu'elle possède un large éventail de fonctionnalités comme des sondages, questionnaire, nuage de mot et remue-méninges, là où les autres applications vont au plus offrir 2 de ses fonctionnalités.

Lorsqu'on compare les sondages de [Digistorm](#) avec ceux de [strawpoll](#) par exemple, [strawpoll](#) a l'avantage d'avoir une meilleure sécurité par rapport aux doubles comptes qui pourrait fausser le sondage, ainsi que la possibilité de changer son vote si on s'est trompé. Cependant, [Digiscreen](#) offre la possibilité de laisser la gestion du sondage à l'animateur.

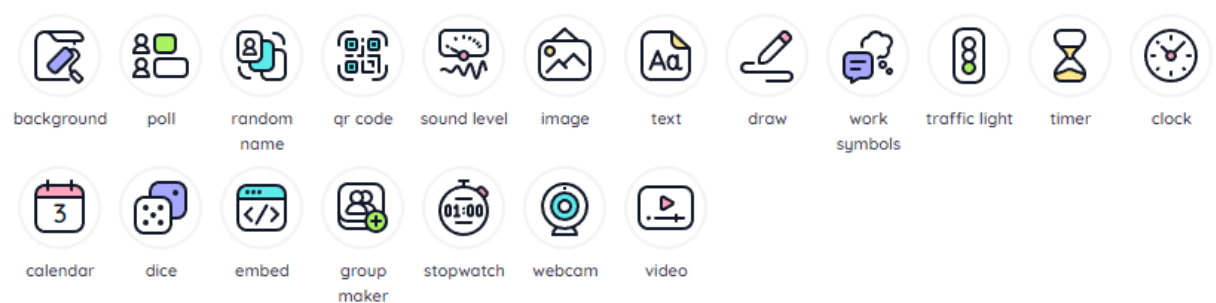
L'expérience utilisateur de [Digistorm](#) n'a rien à envier aux autres applications. Elle possède une interface agréable et intuitive la rendant propice à la réalisation du sondage.

## [Digiscreen](#)

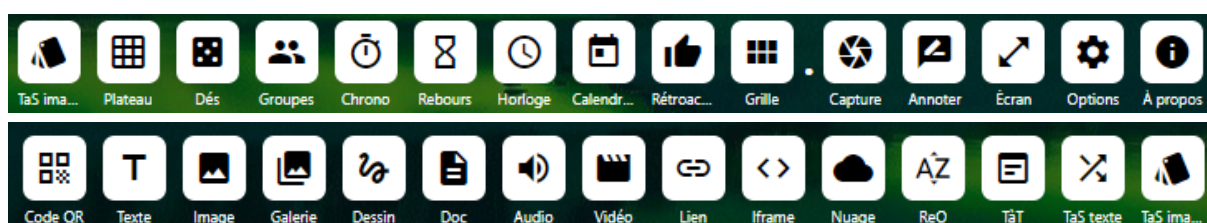
Lorsqu'on compare [Digiscreen](#) avec une application similaire comme [Classroomscreen](#), il possède plus de fonctionnalités que cette dernière (25 contre 19).

Cependant, ce dernier possède des fonctionnalités intéressantes qui ne sont pas implémentées dans [Digiscreen](#) comme l'importation de fond d'écran personnalisé à partir des fichiers de son ordinateur, l'affichage de la webcam ou alors l'intégration d'un sondage, mais cette fonctionnalité est compensé dans l'application [Digiscreen](#).

L'interface de [Classroomscreen](#) permet de cacher certains widgets pour ne pas saturer la barre d'outil ainsi que des logos coloré améliore la lisibilité, donc l'expérience utilisateur est plus agréable.



L'interface de [Digiscreen](#) en revanche est plus exhaustive que celle de [Classroomscreen](#) dû à son manque de couleur ainsi que les widgets rassemblés sur une ligne, ce qui a pour conséquence de réduire la taille de ces derniers ce qui rend le tout moins intuitif.



## Digiboard

### Le cas Openboard

L'avantage de [Digiboard](#) par rapport à d'autres applications similaires comme [Openboard](#) est qu'elle est accessible depuis un navigateur internet, là où [Openboard](#) est une application qui doit être installée sur ordinateur et n'est donc pas compatible avec certains appareils comme les téléphones mobiles. Elle ne peut également être utilisée à plusieurs.

[Digiboard](#) a la particularité de pouvoir insérer des formes comme des rectangles, des cercles ou bien des étoiles contrairement à [Openboard](#). Cependant, [Openboard](#) possède plus de fonctionnalités que [Digiboard](#) comme la possibilité d'avoir plusieurs pages différentes pour conserver des tableaux, là où [Openboard](#) est obligé d'effacer ou bien de changer de tableaux. Il possède aussi l'option de changer de stylo pour écrire ou d'intégrer divers média comme des vidéos ou des sons.

L'expérience utilisateur de [Digiboard](#) est supérieure à celle d'[Openboard](#) dont l'interface utilisateur envahissante gâche de l'espace. De plus, il n'est pas possible d'effectuer du travail collaboratif sur cette dernière.

### Le cas Miro

[Miro](#) est un outil de tableau blanc collaboratif cependant il possède une philosophie différente à celle de [Digiboard](#). En effet, il ressemble plus à un canvas qu'on peut personnaliser comme bon nous semble. Par conséquent, il implémente beaucoup plus de fonctionnalités notamment à l'aide de templates.

Cette personnalisation rend la prise en main beaucoup plus complexe comparé à [Digiboard](#) qui est centré sur le tableau blanc et laisse le reste des applications de la suite [La Digitale](#) couvrir le reste contrairement à [Miro](#) qui essaie d'être tout les outils à la fois, ce qui donne à digiboard une bien meilleure expérience utilisateur.

# Conclusion

Pour synthétiser notre veille technologique, la suite [La Digitale](#) est solution moderne et élégante dont la simplicité rend l'expérience utilisateur intuitive. Elle n'a donc pas besoin de remaniement majeur du back-end ou du front-end, il faudra cependant réaliser deux interfaces pour faciliter l'expérience utilisateur :

- Une interface de connexion client vers serveur en [Kotlin](#).
- Une interface pour démarrer les applications La digitale en [Node.js](#).