

# E-Lycee

Projet final  
Formation en alternance développeur  
L'École Multimédia – Novembre 2014  
**Auteur : Antoine Lucsko**

L'ÉCOLE  
MULTI  
MÉDIA

## Contexte

Les professeurs de mathématiques d'un Lycée d'Ardèche ont décidé de mettre en place un système de QCM automatique en ligne et de créer un site de contenu.

Les élèves se connecteront au site pour accéder à leur espace personnel. Une fois dans celui-ci, ils accéderont à des QCM, aux quels ils répondront. Une fois qu'ils auront fait un QCM, ils auront automatiquement leur score et le QCM changera de statut, il ne sera plus accessible.

Les professeurs se connecteront au site pour créer les QCM et publier des articles. Attention, le CRUD des élèves n'est pas à faire, voir les remarques ci-après.

La partie publique du site présentera le Lycée et affichera des actualités liées aux mathématiques.

Les professeurs veulent mettre en place les sociaux tags Facebook et Twitter sur leur site.

Les enseignants du Lycée fournissent des wireframes pour l'intégration des pages HTML/CSS. Ils donnent quelques pistes graphiques :

<https://ocaml.org/>

<https://www.python.org/>

Le site sera « responsive ».

**Dans la V2 un compte Twitter sera créé, cette partie n'est donc pas à réaliser ici, vous rendrez simplement visible la partie où s'afficheront les tweets dans la sidebar (voir wireframe), de plus le CRUD pour les élèves n'est pas demandé dans la V1, vous aurez un fichier d'élèves de terminale, de première et du professeur à insérer en base de données, dans la table « users ».**

Les seules difficultés techniques sont la création des formulaires pour les questionnaires et l'automatisation des réponses. Le brief vous aidera à trouver les solutions à ces problèmes.

### Fichiers transmis

- Ce brief.
- Liste des élèves de première S et terminale S, et de l'enseignant (fichier JSON).

### Barème sur 200 points

**Le développement du système de QCM est noté sur 50 points (portion technique du sujet). Le reste de l'application est classique et en principe facile à réaliser, en conséquence soignez la première partie pour avoir le maximum de points.**

- Rigueur de la présentation et respect des contraintes techniques
  - Contraintes codes et organisation (Trello) : **20 points.**

- Présentation des données (schéma des tables) : **20 points.**
- Site public administration des articles et commentaires
  - HTML/CSS : **50 points.**
  - Formulaire (sécurité et validation client/serveur) : **20 points.**
  - Framework mise en place : **20 points.**
- Connexion à l'administration  
gestion des « users » avec leurs rôles admin ou « student » : **20 points**
- Implémentation du QCM
  - Dashboard élève **10 points.**
  - Création des QCM **10 points.**
  - Système automatique de score **30 points.**

### Contraintes techniques et outils

- **Intégration**
  - Pour le site public vous pouvez utiliser vos propres CSS.
  - Pour la partie administrable du site vous devez utiliser Twitter Bootstrap.
- **PHP**
  - Framework : Laravel ou Symfony2.
- Vous utiliserez **MySQL** pour la gestion des données
  - Attention, vous devez présenter le schéma des tables lors de votre soutenance.
  - Tous les noms des tables sont au pluriel et les tables de liaisons au singulier.
- **Contrainte pour le CRUDS (Create, Read, Update, Delete, Status )**
  - Les actions suivantes seront accompagnées d'un message de succès ou d'échec :
    - **C** pour la création d'un contenu.
    - **U** pour la mise à jour.
    - **D** est accompagné d'un pop-in de confirmation qui doit être validé avant de réaliser l'action de suppression, suivi d'un message de succès ou d'échec, qui s'affiche et s'efface au bout de quelques secondes (2 ou 3).
    - **S** pour changer le statut d'un contenu (publié/dépublié).
    - **R** permet de lire le contenu d'un item.
    - Pour les actions **C, S** et **U**, à chaque fois qu'elles sont réalisées vous êtes redirigé vers la page d'administration du CRUD en question, et un message de succès ou d'échec s'affiche.
- Pas de CRUD pour les commentaires.
- Toutes les dates affichées sur le site public ou dans l'espace d'administration seront standardisées comme suit :
  - **JJ MM AAAA**
- Tous les champs des formulaires obligatoires seront mentionnés, et les données utilisateurs seront vérifiées côté client et serveur.

- Le code ne doit pas être trop commenté et indenté correctement (voir <http://www.php-fig.org/>).
- Vous utiliserez Trello ou un autre système pour planifier le développement entre vous.

*Remarques :*

*Vous pouvez également utiliser GRUNT, Git ...*

## **Rendu/ livrable**

Vous devez rendre un site livrable et facilement déployable sur un serveur de production. Code de l'application et données d'exemple.

## **Schéma des données**

Vous partirez des « schémas » ci-dessous pour implémenter le code SQL de vos tables :

### **users**

id
username
password
role: <b>teacher,</b> <b>first_class,</b> <b>final_class</b>

### **posts**

id
user_id
title
abstract
content
url_thumbnail
date
status

### **comments**

id
title
content
date
status

## questions

id
title
content
class_level : <b>terminale, premiere</b>
status : <b>publish,</b> <b>unpublish</b>

## choices

id
question_id
content
status:yes, no

## scores

id
user_id
question_id
status_question : <b>fait, pas fait</b>
note

## Objectif du site public

Vous devez réaliser les pages suivantes :

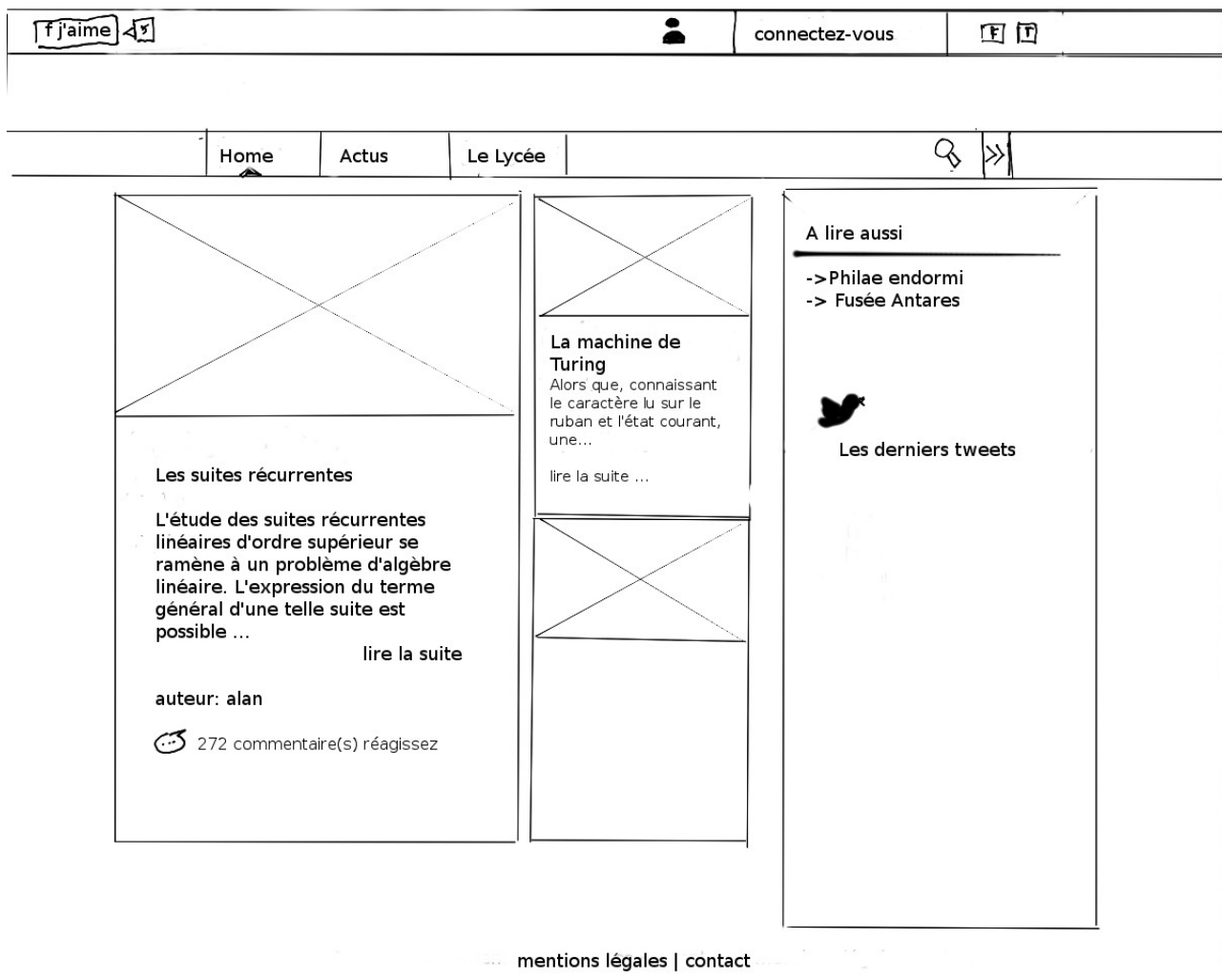
- Page d'accueil.
- Actus (toutes les actualités).
  - Page d'une actualité.
- Le Lycée (présentation du Lycée page statique).
- Mentions légales et contact (pages statiques).
- Connexion à l'espace privé.

## Directives

- **Page d'accueil (voir wireframe ci-dessous)**
  - Menu haut
    - Un bouton « j'aime » Facebook.
    - Un lien vers la connexion à l'espace privé, pour les élèves et professeurs. Si une personne est connectée, vous placerez un bouton pour aller rapidement à l'espace d'administration depuis cette barre. Elle sera visible sur toute les pages du site.
    - Des « socials tags » (Facebook et Twitter).
  - Menu principal
    - Home

- Actus
  - Lycée
  - Un moteur de recherche sur les articles du site.
- Contenu (colonne centrale)
  - Les articles seront présentés comme dans le wireframe ci-dessous.
    - Chaque bloc article contiendra
      - Une image à la une
      - Un titre
      - Un extrait + lire la suite
      - L'auteur et la date de publication
      - Le nombre de commentaire(s)
- Sidebar (colonne latérale)
  - Une liste de titre d'articles supplémentaires du site cliquables dans une catégorie « à lire aussi ».
  - Les derniers tweets du compte du lycée (**pas fonctionnel dans la V1**).
- Pied de page
  - Un lien vers la page « mentions légales ».
  - Un lien vers la page contact du site.

### Wireframe page d'accueil ci-dessous :



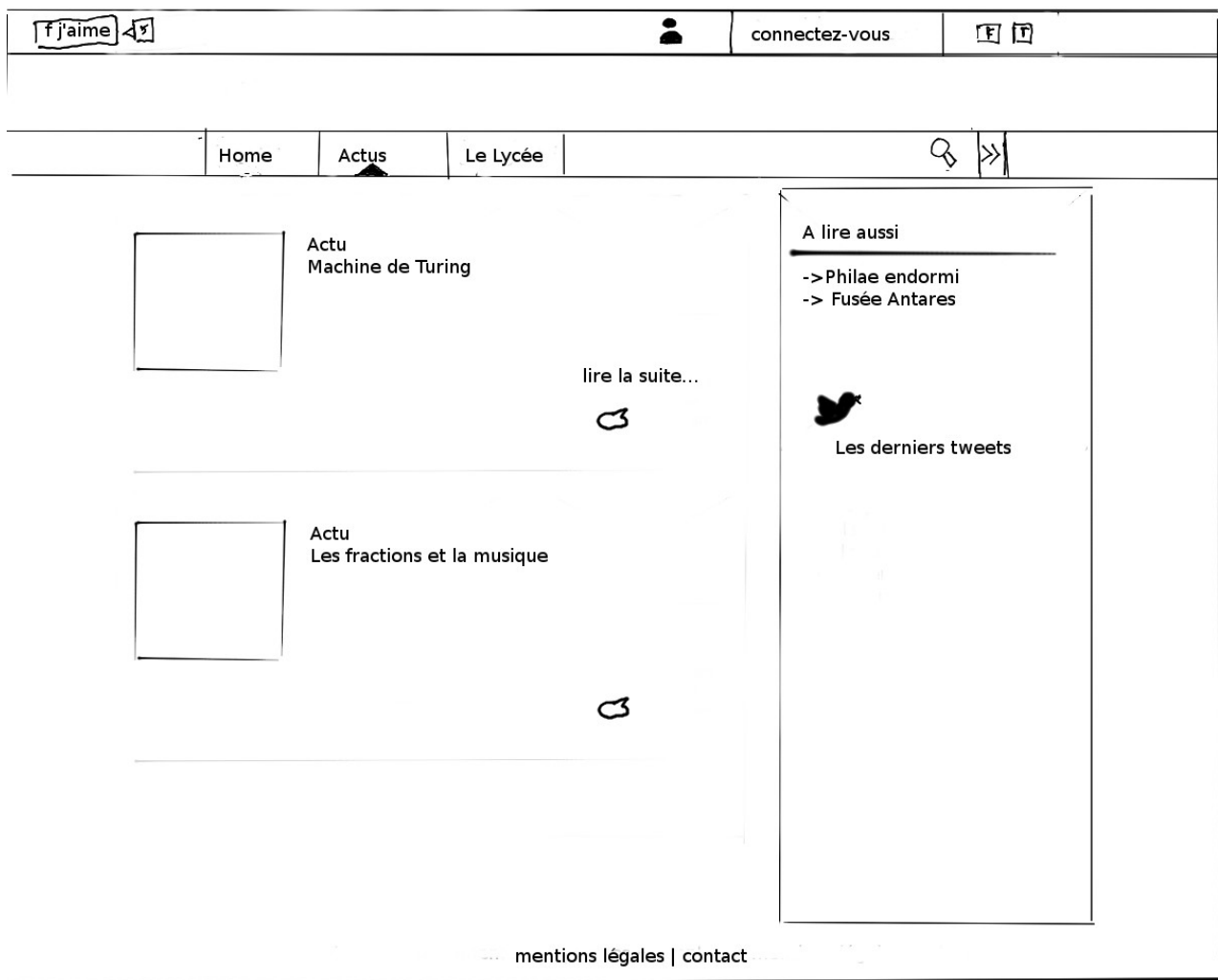
- **Page des actualités et une page d'une actualité**

La page d'actualités présentera sous forme d'une liste les actualités par ordre de date de publication.

Chaque item de cette liste contiendra :

- Une image à la une.
- Un titre .
- Un extrait + lire la suite.
- L'auteur et la date de publication.
- Le nombre de commentaire(s).

**Wireframe actualités :**



- **Page d'une actualité**

Dans la page d'une actualité vous afficherez le contenu de l'article ainsi que la possibilité d'écrire un commentaire lié à l'article. Vous utiliserez un filtre de spam comme Akismet.

- **Page de présentation du Lycée, de contact et mentions légales**

Vous reprendrez la structure d'une page d'une actualité et l'adapterez pour ces contenus.

- **Page de contact**

Pensez à la sécurité pour les formulaires. La page de contact enverra un e-mail au propriétaire du site.

### Wireframe page de contact :

<span>f j'aime</span>			<a href="#">connectez-vous</a>	
<a href="#">Home</a>		<a href="#">Actus</a>	<a href="#">Le Lycée</a>	

#### contact Lycée

e-mail

sujet

commentaires

[envoyer](#)

#### A lire aussi

- > Philae endormi
- > Fusée Antares

Les derniers tweets

[mentions légales](#) | [contact](#)



- **Page de connexion**

Vous sécuriserez les données utilisateur ; pour se connecter à l'administration du site, il faut un identifiant et un mot de passe. Vous n'implémenterez pas l'option « Mot de passe oublié ».

A hand-drawn sketch of a login form. It features two input fields: the first is labeled 'identifiant' and the second is labeled 'Mot de passe'. Below the password field, there is a checkbox labeled 'Se souvenir de moi' and a button labeled 'connexion'. At the bottom of the form, there is a link 'Mot de passe oublié ?' and a link '<- retour sur le site'.

### **Site privé, administration pour les enseignants.**

Vous utiliserez Twitter Bootstrap. Les pages d'administration sont toutes sur deux colonnes

- Sidebar à gauche.
- Une colonne centrale.

Vous devez réaliser uniquement les 2 CRUDS suivants

- CRUDS des articles.
- CRUDS des fiches élèves.

- **Page dashboard**
  - **Menu haut**
    - Un lien vers le site public « Retour au site public ».
    - Une icône indiquant la personne connectée (nom du professeur).
    - Un lien pour se déconnecter.
  - **Menu sidebar (colonne latérale droite)**
    - Un bouton « dashboard indiquera que nous sommes sur cette page.
    - Un lien vers le CRDS des fiches.
    - Un lien vers le CRUDS des articles.
    - Un lien vers le CRUDS des élèves ( **n'est pas à réaliser dans la V1**).
  - **Colonne centrale**
    - Trois onglets dépliant et listant les 3 derniers contenus publiés ou dépubliés avec leur statut respectif en couleur, vert publié et rouge dépublié.
    - Un onglet statistique
      - Le nombre de commentaires publiés sur le site.
      - Le nombre de fiches publiées.
      - Le nombre d'élèves.

**Wireframe dashboard enseignants ci-dessous :**



- **CRUDS des articles page principale listant les articles**

- Titres (items) cliquables.
  - Le lien mènera vers un formulaire de mise à jour, fonction de la table posts.
- Un bouton pour ajouter un contenu
  - Le lien mènera vers un formulaire de création, fonction de la table posts.
- Un système de pagination.
- Action groupée pour la publication, dépublication et suppression (attention la suppression doit être accompagnée d'un message de confirmation type pop-in).

**Wireframe CRUDS ci-dessous :**

retour au site public

hello admin  
se déconnecter

dashboard

fiches

articles

commentaires

pages

élèves

Articles

Ajouter

publié

Appliquer

<input type="checkbox"/>	titre	Auteur		statut
<input type="checkbox"/>	La machine de Turing	Alan	0	
<input type="checkbox"/>	Les suites numériques	Henri		
<input type="checkbox"/>	Javascript	Thierry		

## CRUD des fiches

**La page principale du CRUDS est identique à celle concernant les articles. Elle présente l'ensemble des QCM.**

### Création d'une fiche

- Elle se fera en 2 étapes
  - Un premier formulaire déterminera la question à poser et le nombre de choix possible pour cette question ainsi que la classe concernée (Terminale S ou première S).
  - Un deuxième formulaire permettra de saisir le texte des questions posées, ainsi que leur valeur vraie ou fausse dans le QCM.

**Une fois le QCM créé, vous ne pouvez plus ajouter des champs à celui-ci. Vous pouvez cependant, dans le CRUD, modifier le contenu de ce formulaire.**

Lorsque la fiche est créée, **il faudra la publier pour que l'ensemble des élèves de la classe de terminale ou de première puisse répondre aux questions.** Il n'est pas demandé de faire des statistiques sur les réponses des élèves.

retour au site public

hello admin

se déconnecter

dashboard

fiches

articles

commentaires

pages

élèves

Ajouter

étape 1/2

niveau

Terminale S

▼

Nombre de choix (\*)

3

titre (\*)

Rédaction de la question (\*)

La machine de Turing c'est quoi ?

envoyer

retour au site public

hello admin  
se déconnecter

dashboard

fiches

articles

commentaires

pages

élèves

Ajouter

étape 2/2

Question 1

La machine de Turing est une super machine à laver

☐ oui
☒ non

Question 2

La machine de turing est une machine qui sert à voyager dans le temps

☐ oui
☒ non

Question 3

La machine de Turing est un modèle abstrait du fonctionnement des appareils mécaniques de calcul

☒ oui
☐ non

envoyer

## Remarques

Il faudra trouver un « moyen » pour informer les élèves qu'un nouveau QCM est à réaliser. Vous pouvez par exemple utiliser la table « scores », une fois le QCM créé. Cette table peut en effet « récupérer » l'id de la question et l'id des utilisateurs de terminale ou de première. Par exemple si vous créez le QCM d'identifiant 1 vous pouvez insérer les données suivantes dans la table « scores ».

Id	user_id	question_id	status	note
1	1	1	Pas fait	0
2	2	1	Pas fait	0

## Partie privée pour les élèves

Une fois connectés les élèves sont redirigés vers leur administration simplifiée.

- **Page dashboard**

- **Menu haut**

- Un lien vers le site public « Retour au site public »
    - Une icône indiquant la personne connectée (nom de l'élève)
    - Un lien pour se déconnecter

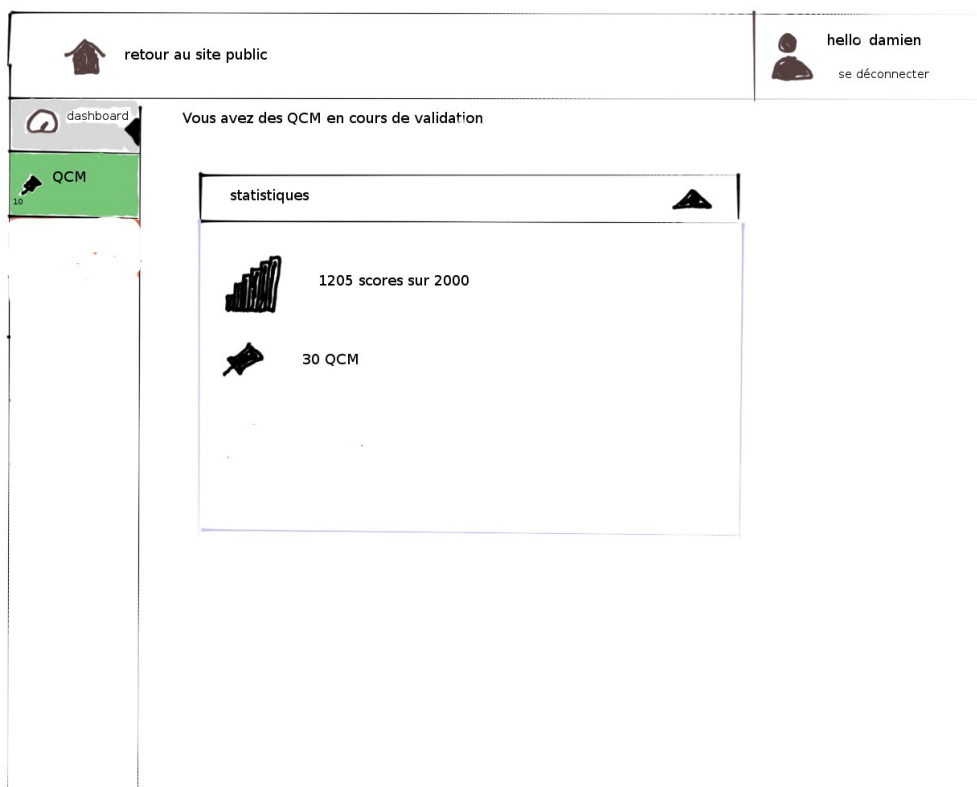
- **Menu sidebar (colonne latérale droite)**

- Un bouton « dashboard indiquera que l'on est sur cette page
    - Un lien vers les QCM à faire

- **Colonne centrale**

- Un onglet statistique
      - Nombre de QCM réalisés
      - Score

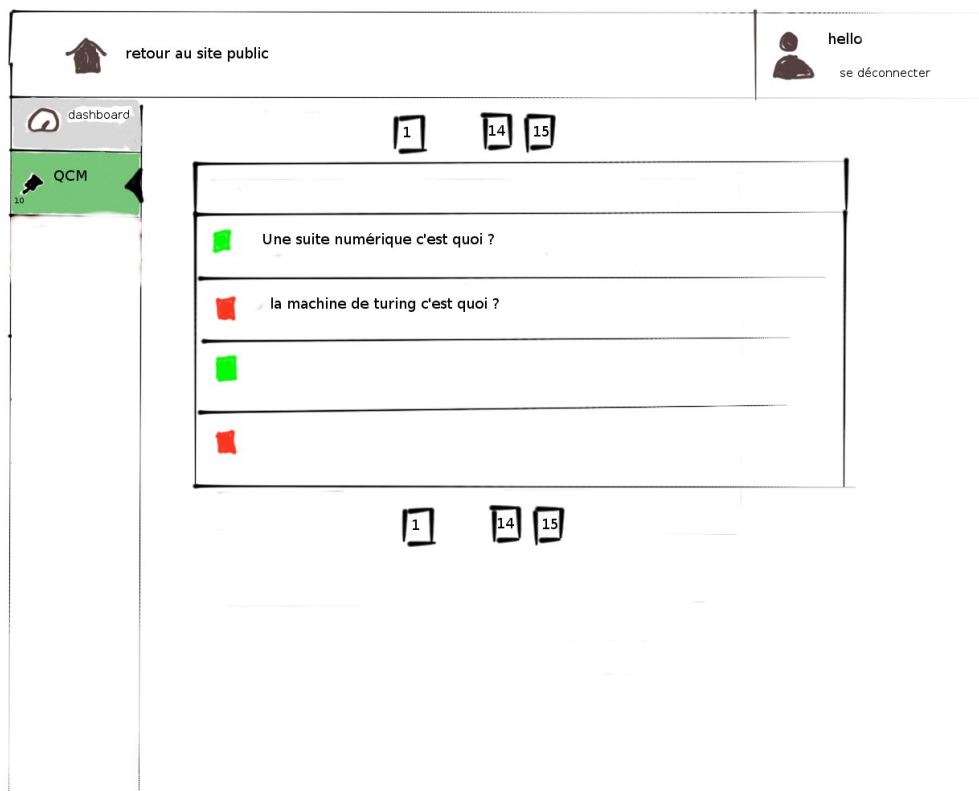
## Wireframe dashboard pour un élève



## Page QCM

La page QCM présente la liste des QCM ; si un QCM est terminé il n'est plus cliquable. Vous utiliserez une couleur pour préciser si le QCM est fait ou pas.

### Wireframe liste des QCM





## Page d'un QCM


Une fois que l'élève a répondu au QCM, il ne sera plus accessible, mais présent dans la liste, et son score sera calculé.


### Wireframe d'un QCM ci-dessous



 retour au site public

 hello damien  
se déconnecter

 dashboard

 QCM

La machine de Turing, c'est quoi ?

C'est une machine à laver

oui ☐ non ☐

C'est une machine qui sert à voyager dans le temps ?

oui ☐ non ☐

Une machine de Turing est un modèle abstrait du fonctionnement des appareils mécaniques de calcul

oui ☐ non ☐

envoyer