# Code WebDeb

Pour obtenir et modifier le code source de WebDeb, il faut installer Git :

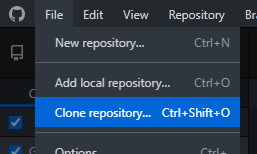
<https://git-scm.com/>

Et ensuite le logiciel GitHub Desktop:

<https://desktop.github.com/>

Une fois le logiciel installé :

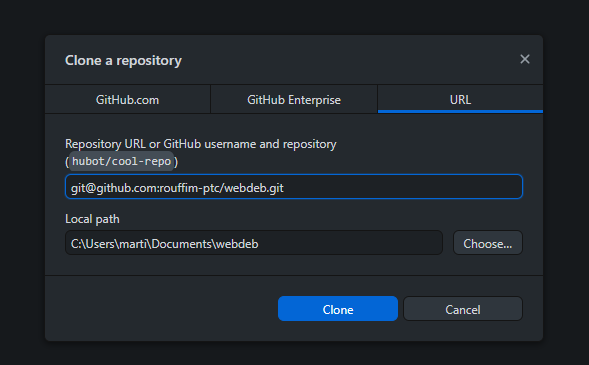
File -> Clone repository



Ensuite onglet URL :

Repository URL : [git@github.com:rouffim-ptc/webdeb.git](mailto:git@github.com:rouffim-ptc/webdeb.git)

Local path : Le dossier où le code sera situé sur l’ordinateur

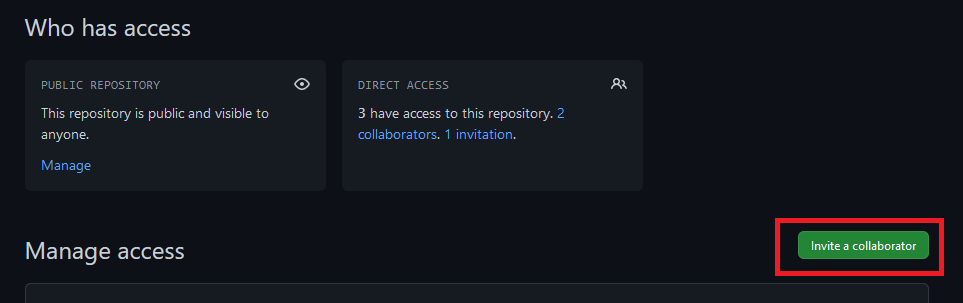


Ensuite appuyer sur clone.

Pour pouvoir apporter des modifications au code, il faut avoir accès au repository : <https://github.com/rouffim-ptc/webdeb>.

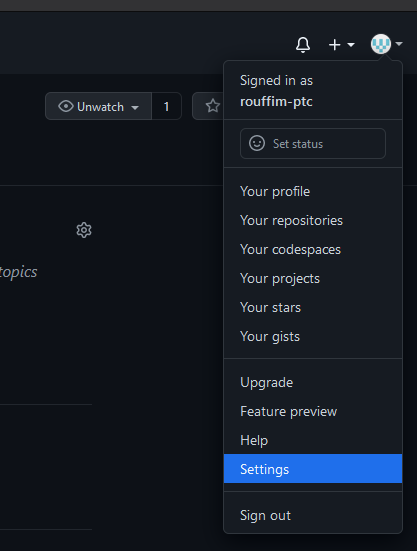
Pour ajouter un accès, seul l’owner du repository peut le faire. L’owner doit se rendre sur la page <https://github.com/rouffim-ptc/webdeb/settings/access>

Ensuite appuyer sur « Invite a collaborator »

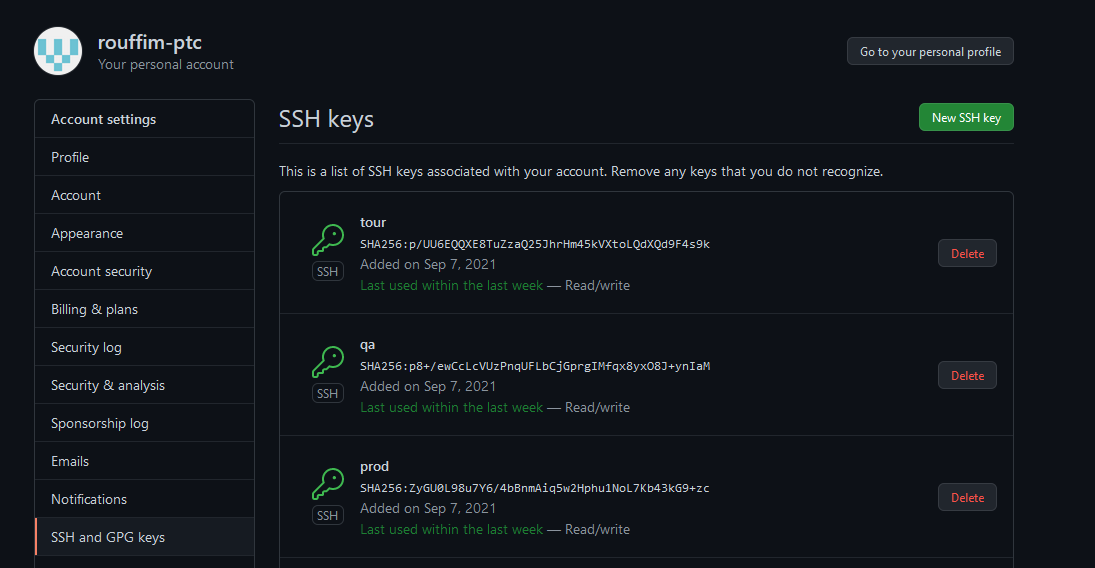


Une fois l’accès obtenu, aller sur <https://github.com/rouffim-ptc/webdeb>.

Appuyer sur le bouton en haut à droite, puis Settings.

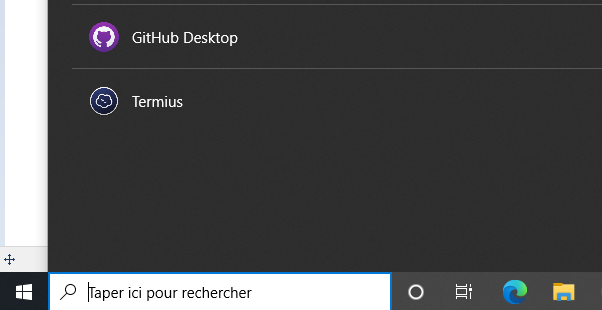


Ensuite aller sur l’onglet SSh and GPG keys, puis cliquer sur New SSH Key



Il faut alors créer une clé SSH sur l’ordinateur. Sur Windows :

Dans la recherche, taper cmd.

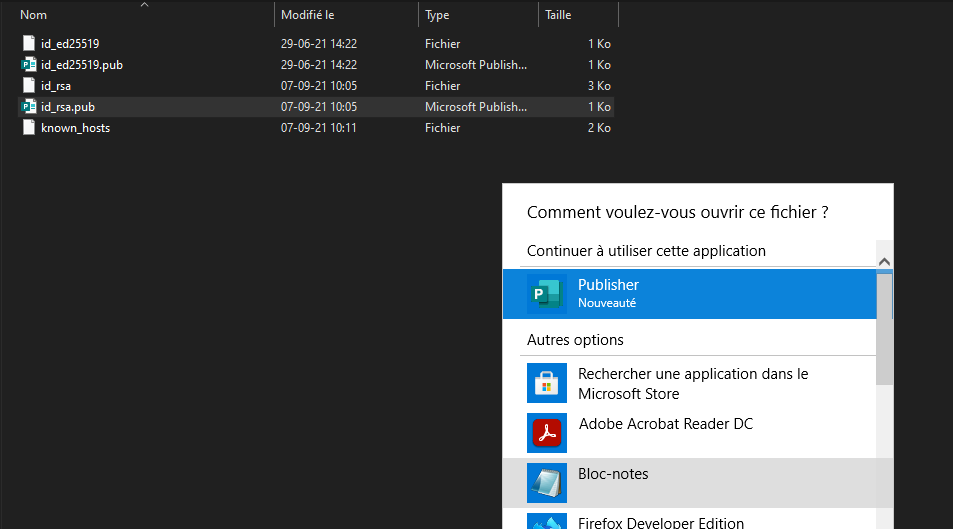


Cela va ouvrir la console. Ensuite introduire la commande : ssh-keygen. Appuyer sur Enter à chaque question posée.

Dans la console, un path est indiqué du style : (C:\Users\marti/.ssh/id\_rsa)

Il faut se rendre dans : C:\Users\marti/.ssh.

Faire un clique droit sur le fichier id\_rsa.pub, puis ouvrir avec Bloc-notes.

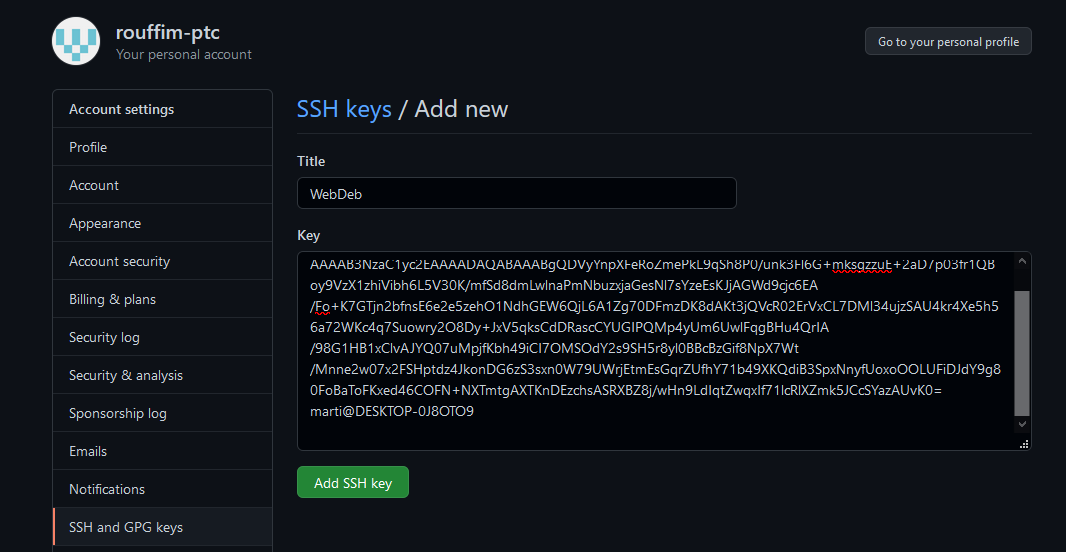


Copier le contenu de ce fichier puis retourner sur github.

Si ce n’est déjà fait, cliquer sur New SSH key.

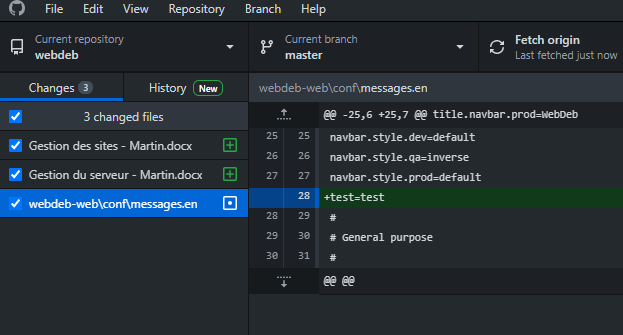
Title : WebDeb

Key : coller le contenu du fichier id\_rsa.pub



Ensuite Add SSH key.

Ensuite, quand un fichier du code WebDeb est modifié, il sera affiché dans GitHub Desktop :

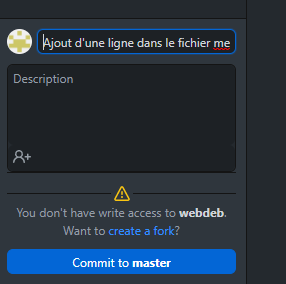


On peut voir qu’une ligne a été ajouté au fichier messages.en.

Ensuite pour pousser les modifications sur le repository central de WebDeb :

Summary : Expliquer ce que la modification du code apporte

Description : laisser vide



Ensuite appuyer sur Commit to master.

# Exportations et référencements

Modifier les textes pour le partage sur les moteurs de recherche, Facebook, mail, …

Les messages se situent dans les fichiers messages.fr, .en, .nl

Les clés sont :

**general.metatitle.title.org.title**= **org pour organisation, metatitle.title est le titre de la page (celui qui apparait sur les moteurs de recherches).   
general.metatitle.title.org.desc**=Même chose mais .desc signifie que c’est la description, c’est-à-dire ce qui peut apparaitre en dessous du titre.

**general.metatitle.og.org.title**= **metatitle.og titre de la page sur les réseaux sociaux   
general.metatitle.og.org.desc**=Même chose mais .desc signifie que c’est la description, c’est-à-dire ce qui peut apparaitre en dessous du titre.

={0} signifie qu’on affiche une données envoyées par le programme. Par exemple le nom d’un acteur.

Pour les mails : **media.mail.subject**

# Aide

Pour ajouter une nouvelle fiche aide, il faut aller dans le fichier :

<https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/app/be/webdeb/presentation/web/controllers/Application.java>

Ensuite chercher la méthode : List<String> getHelpList()

Il suffit alors de modifier le tableau values pour modifier les fiches aides. L’ordre du tableau est l’ordre d’apparition des fiches.

String values [] = {

"entry.actor",

"group",

"help.subscribe"

};

Ensuite trois valeurs possible pour la clé :

**help.title. pour le titre de la fiche**

**help.manual. pour le nom du fichier pdf (optionnel). Attention, le nom du fichier ne doit pas contenir de caractères spéciaux ou d’espaces.**

**help.video**. pour le code de la video youtube (optionnel).

Si le lien est : <https://www.youtube.com/watch?v=MhZDYjQCl50>

Le code est : MhZDYjQCl50

Par exemple pour l’acteur on aura :

help.title.entry.actor

help.manual.entry.actor

help.video.entry.actor

Les fichiers pdf sont à ajouter dans les dossiers :

PDF en anglais :

[/webdeb-sources/src/master/webdeb-web/public/static/rsc/manuals/en/](https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/public/static/rsc/manuals/en/)

PDF en français :

[/webdeb-sources/src/master/webdeb-web/public/static/rsc/manuals/fr/](https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/public/static/rsc/manuals/fr/)

PDF en néerlandais

[/webdeb-sources/src/master/webdeb-web/public/static/rsc/manuals/nl/](https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/public/static/rsc/manuals/nl/)

# Bulle d’aide

Les bulles d’aides peuvent être modifiée dans les différents formulaires. Voici la liste :

Acteur :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/actor/editActorFields.scala.html

Fonction :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/actor/editProfession.scala.html

Argument :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/argument/editArgumentFields.scala.html

Citation :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/citation/editCitationFields.scala.html

Débat :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/debate/editDebateFields.scala.html

Dossier :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/tag/editTagFields.scala.html

Texte :

/webdeb-web/app/be/webdeb/presentation/web/views/entry/text/editTextFields.scala.html

et

/webdeb-web/app/be/webdeb/presentation/web/views/entry/text/editTextMainFields.scala.html

Contributeur inscription :

/webdeb-web/app/be/webdeb/presentation/web/views/account/settings/contributorFormOnly.scala.html

Contributeur login :

/webdeb-web/app/be/webdeb/presentation/web/views/account/settings/loginFormOnly.scala.html

Uniquement pour les champs de type :

**@**textInput(

**@**urlInput(

**@**textareaInput(

**@**selectInput(

**@**emailInput(

Il doit y avoir absolument 6 paramètres avant la bulle d’aide. C’est-à-dire par exemple pour un champ comme ceci :

**@**textInput(  
 actorForm(**"name.last"**),  
 *Messages*(**"actor.label.lastname"**),  
)

Il faut donc ajouter dans ce cas 4 paramètres pour ajouter la bulle :

**@**textInput(  
 actorForm(**"name.last"**),  
 *Messages*(**"actor.label.lastname"**),

null, null, null, null,

helpbubble(**"bubble.title"**, **"bubble.entry.actor.pseudo"**),  
)

SI il y a un “false”, il faut le laisser!

où bubble.title est le titre de la bulle

et **bubble.entry.actor.pseudo est le contenu**

**Pour changer ce contenu il suffit d’aller le fichier message et changer le titre correspondant.**

**Pour ajouter un nouvel item, ajouter la clé dans le fichier messages ex :**

**bubble.other.title=Nouveau titre**

Pour aller à la ligne, il faut utiliser du HTML (UNIQUEMENT POUR LES BUBBLES !)

Ex :

bubble.citation.content.extraitoriginal=Contenu de la bulle de test.<br> \

Pas super original.

Mettre un null en moins pour :

**@**radioInput(

**@**checkboxInput(

**@**passwordInput(

**@**colorInput(

Attention, il faut rajouter la bulle 1 seule fois dans le code sur bitbucket, mais une fois dans chaque document langue (1x fr, 1x nl, 1x en)

# Page enseigner

[/webdeb-web/app/be/webdeb/presentation/web/views/teach.scala.html](https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/app/be/webdeb/presentation/web/views/teach.scala.html?at=master)

Le design fonctionne avec Bootstrap 4.

Si des fichiers css complémentaire pour le design de la page les ajouter ici :

[/webdeb-web/public/stylesheets/design/](https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/public/stylesheets/design/?at=master)

Ensuite ajouter les css dans la page teach.scala.html et les mettre entre les {} de

@scripts = {

<link rel="stylesheet" media="screen" href="@routes.Assets.at("stylesheets/design/nom\_du\_fichier.css")">

}

Ensuite ajouter le contenu html entre les {} de:

**@**main(user, scripts, *Html*(**""**), **""**, **"viz.teach"**) {

Code html ici

}

# Ajouter une video explorer

<https://bitbucket.org/webdeb/webdeb-sources/src/master/webdeb-web/app/be/webdeb/presentation/web/views/viz/explore.scala.html?at=master>

Sur youtube, appuyer sur partager la vidéo, puis intégrer. Copier ensuite le code <iframe></iframe> à l’endroit souhaiter. Les paramètres width="560" height="315" peuvent être modifié afin de changer la taille de la vidéo.

# Ajouter des PDF pour Débagora

Les fichiers pdf sont à ajouter dans les dossiers :

PDF en anglais :

webdeb\webdeb-web\public\static\rsc\debagora\en

PDF en français :

webdeb\webdeb-web\public\static\rsc\debagora\fr

PDF en néerlandais

webdeb\webdeb-web\public\static\rsc\debagora\nl

Ensuite mettre à jour le fichier webdeb\webdeb-web\app\be\webdeb\presentation\web\views\others\debagora.scala.html

Le code des cards Debagora ressemble à ceci :

<div class="card col-12 col-md-6 col-lg-3 module2">  
 <div class="module-image-container">  
 <div class="module-image" style="background-image: url(assets/images/others/debagora/module2.jpg)"></div>  
 </div>  
 <div class="card-body">  
 <h5 class="card-title">  
 Module 2 — Faire émerger et formuler une question de débat  
 </h5>  
 <div class="card-text">  
 <h6>Ce module propose des activités pour :</h6>  
 <ul>  
 <li>faire émerger des sujets de préoccupation et d’interrogation ;</li>  
 <li>transformer des préoccupations en questions politiques, et comprendre ainsi que le politique n’est pas un concept abstrait et touche la vie concrète ;</li>  
 <li>formuler une question de débat claire et précise.</li>  
 </ul>  
 </div>  
 <a href="@routes.Assets.at("static/rsc/debagora/" + ctx.lang.code + "/debagora.pdf")" class="stretched-link"></a>  
 </div>  
</div>

Il suffit alors de modifier le dernier lien, @routes.Assets.at

Remplacer debagora.pdf par le nom du pdf (éviter les espaces et caractères spéciaux dans le nom de fichier).