



Aller sur l'ENT de l'UPEM...

Aller dans « Mes sites web »

The screenshot shows the UPEM ENT website interface. At the top, there is a navigation bar with the UPEM logo and the text 'ENT'. Below this, there is a main navigation menu with various categories like 'Ma formation', 'Vie quotidienne', etc. A dropdown menu is open under 'Informations pratiques', showing options like 'Guides', 'Plans', 'Contacts', and 'Mes services informatiques'. The 'Mes services informatiques' option is expanded, showing a list of services including 'Mes pages WEB', 'Mes bases de données', 'Mes listes de diffusion', 'Mes fichiers', and 'Mes impressions en bibliothèque'. The 'Mes pages WEB' option is highlighted with a mouse cursor. In the background, there is a large white box with the text 'Bienvenue sur l'Espace Numérique de Travail' and a smaller box with the text 'Veillez vous authentifier...'. The browser's address bar shows the URL 'https://ent.u-pem.fr/Portal/f/u8601s700/normal/render.uP'. The browser's taskbar at the bottom shows several open applications and the system clock at 17:40.

Accueil | UPEM - Mozilla Firefox

Accueil | UPEM Services FormationSup Principaux diplômes et for... Untitled project, Page1 HTML5 : structure globale ch #254b46 Color Hex CSS Buttons

https://ent.u-pem.fr/Portal/f/u8601s700/normal/render.uP

UP EM ENT

Personnaliser

PEREIRA TONY

Accueil

Ma formation Vie quotidienne Orientation et insertion professionnelle Partir à l'étranger Vie associative et culturelle Réinscription et candidature Informations pratiques

UP EM

Guides Plans Contacts Mes services informatiques

- Mes pages WEB
- Mes bases de données
- Mes listes de diffusion
- Mes fichiers
- Mes impressions en bibliothèque

Mes pages WEB

Veillez vous authentifier... ont été remis pour prendre l'environnement.

Pour ce faire cliquez sur

Si vous ne disposez pas de vos codes d'accès, cliquez sur la catégorie qui vous concerne :

- Personnel : contactez le CRI au 01 60 95 74 55
- Etudiant

Bienvenue sur l'Espace Numérique de Travail

L'université Paris-Est Marne-la-Vallée met à la disposition de ses personnels et de ses étudiants un Environnement Numérique de Travail (ENT) ou bureau à distance.

Celui-ci permet, à partir d'un navigateur, d'accéder aux services mis à disposition, depuis n'importe quel lieu et à n'importe quel moment. Il suffit de disposer d'un équipement connecté à l'internet (depuis son domicile, depuis le réseau de l'université ou depuis tout autre point d'accès).

Pour accéder à l'ENT, vous devez être régulièrement inscrit en qualité d'étudiant ou bien faire partie du personnel de l'établissement, et disposer d'un identifiant et du mot de passe associé.

L'ENT vous permet d'accéder à :

- votre messagerie électronique
- vos fichiers personnels
- vos pages web,...

Espace Numérique de Travail

https://www.u-pem.fr/espace-numerique-de-travail-etudiants/informations-pratiques/mes-services-informatiques/mes-pages-web/

Menu Accueil | UPEM - Mo [descriptif doclect... jstyle.css -- /WWW... WWW-perso [Dossier personnel] [Sans titre]1.0 (Co... 17:40

Activer son site web, puis vérifier que ça marche

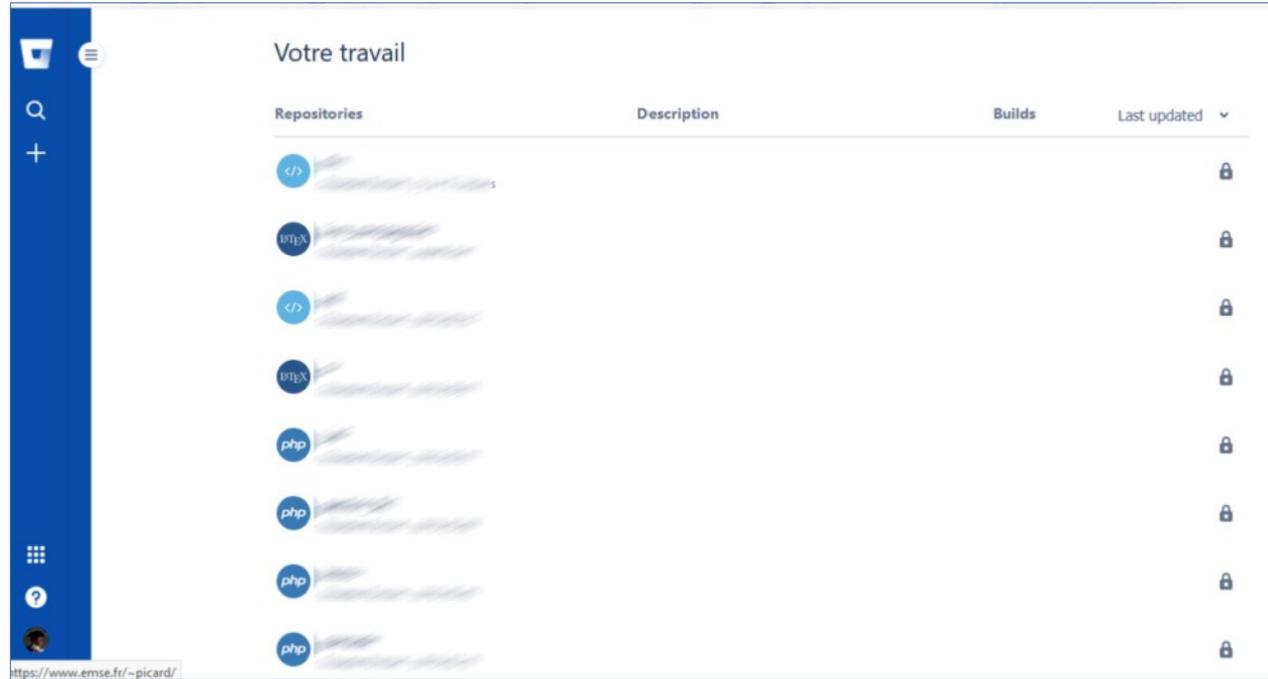
A LIRE	PAGES WEB PÉDAGOGIQUES	PAGES WEB PERSO	Site web activé. Vos pages web sont publiées. Elles sont visibles à l'adresse : https://perso-etudiant.u-pem.fr/~[redacted]
PRÉSENTATION DU SERVICE	ACTIVER MON SITE WEB	ACTIVER MON SITE WEB	
	VOIR MON SITE WEB	DÉSACTIVER MON SITE WEB	
		VOIR MON SITE WEB	



<https://bitbucket.org/>

Bitbucket est une « forge web pour auto-hébergement du code »
Il en existe d'autres, comme github.

Créer son compte Bitbucket : <https://bitbucket.org/>



2. Créer un dépôt « opendata »

The image shows a screenshot of the Bitbucket web interface. On the left, a vertical blue sidebar contains a search icon and a plus sign (+) which is circled in red. An arrow points from this plus sign to the right-hand side of the image, which is a zoomed-in view of the 'Créer un dépôt' (Create repository) menu. In this menu, the 'Dépôt' (Repository) option is circled in red. The menu is titled 'Bitbucket' and includes a back arrow. Under the 'CREATE' section, there are five options: 'Dépôt' (with a code icon), 'Équipe' (with a group icon), 'Projet' (with a folder icon), and 'Snippet' (with a scissors icon). Under the 'IMPORT' section, there is one option: 'Dépôt' (with a code icon). The background of the main interface shows a table with columns 'Repositories', 'Description', and 'Builds', and a list of repository entries with icons for various languages like PHP and Python.

Créer un dépôt

[Importer un dépôt](#)

Nom de dépôt*

opendata

Niveau d'accès

Ceci est un dépôt privé

Uncheck to make this repository public. Public repositories typically contain open-source code and can be viewed by anyone.

Inclure un fichier
LISEZMOI ?

No

Système de contrôle
de version

Git

Mercurial

▼ Paramètres avancés

Description

Faire un fork

Aucun fork

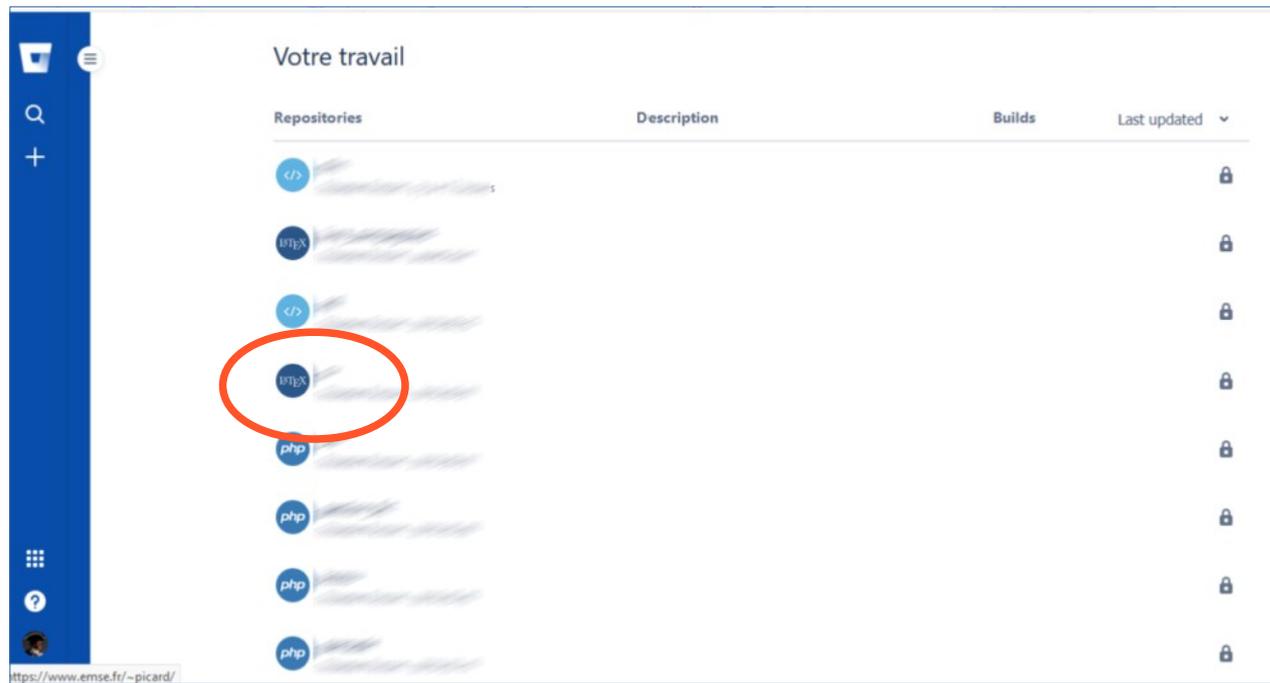
Langue

PHP

Créer un dépôt

[Annuler](#)

2. Créer un dépôt « opendata »



The screenshot shows a user's profile page titled "Votre travail" (Your work). It features a table of repositories with columns for "Repositories", "Description", "Builds", and "Last updated". The "Repositories" column contains icons for different languages: C#, PHP, C#, and a circled PHP icon. The "Last updated" column shows lock icons for each repository. The URL at the bottom left is <https://www.emse.fr/~picard/>.

Repositories	Description	Builds	Last updated
	[blurred]		🔒
	[blurred]		🔒
	[blurred]		🔒
	[blurred]		🔒
	[blurred]		🔒
	[blurred]		🔒
	[blurred]		🔒
	[blurred]		🔒

3. Copier la ligne de commande pour cloner le dépôt



Placez quelques bits dans votre paquet



Lancez-vous

Créer un fichier LISEZMOI ou .gitignore est un moyen rapide et facile de placer quelque chose dans votre dépôt.

Créer un fichier LISEZMOI

Créer un .gitignore



git



Git est un logiciel de gestion de versions décentralisé.

C'est un logiciel libre créé par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2.

En 2016, il s'agit du logiciel de gestion de versions le plus populaire qui est utilisé par plus de douze millions de personnes.

1. Dans votre terminal linux, aller dans votre espace web WWW de l'upem
2. Coller la ligne de commande précédente (pour cloner votre dépôt)
→ Ceci clone votre dépôt.
3. Afficher tous les fichiers/dossiers de WWW/ (**ls -la**)
→ vous voyez un nouveau dossier « opendata » dans WWW/
4. Aller dans opendata/ (**cd opendata**)
5. Créer un fichier « index.html » (**vi index.html**)
6. Insérer « Hello world », enregistrer et quittez le fichier
(i → Hello world → ECHAP :wq ENTRER)
7. Demander le statut du dépôt (**git status**)
→ il vous indique la présence d'un nouveau fichier non ajouté au dépôt
8. Ajouter le fichier au dépôt (**git add index.html**)

9. On considère que c'est une version fonctionnelle intéressante, on va commiter (« valider une transaction ») le code actuel du dépôt :
git commit -am"version' initiale"

10. On considère qu'on peut pousser notre code sur le dépôt :
git push

11. Rafraîchir le code source du dépôt dans bitbucket
→ on voit désormais le fichier, hourra !

12. Modifier le fichier « index.html »

13. Faire un **git statuts** et voir ce que git répond

14. Faire un **git diff** et voir ce que git répond

15. Commiter et pousser le code

16. Rafraîchir le code source du dépôt dans bitbucket

17. Vous voyez désormais l'historique de votre code et pouvez les consulter :

The screenshot shows the Bitbucket interface for a repository. At the top, the file name 'index.php' is displayed. Below it, a message states: 'C'est ici que vous trouverez les fichiers sources du dépôt. Si vous souhaitez donner aux utilisateurs une idée de description à votre dépôt.' The interface includes a 'Source' dropdown menu, a 'master' branch selector, and a commit hash 'a94baf5' which is circled in red. A 'Full commit' button is also visible. The main content area shows the PHP code for 'index.php'. A 'COMMIT HISTORY' dropdown menu is open, also circled in red, showing a list of commits. The first commit is highlighted in blue and contains the text 'bug fix' and 'avant-hier'. Other commits include 'update website' (il y a 3 jours), 'nouvelle version site' (2019-09-22), 'html5 validation ok' (2018-08-27), and another commit (2018-07-05). The code in the background includes PHP tags for require_once, exit, and HTML meta tags for charset, http-equiv, viewport, and description.

```
index.php

C'est ici que vous trouverez les fichiers sources du dépôt. Si vous souhaitez donner aux utilisateurs une idée de description à votre dépôt.

Source  master  a94baf5  Full commit

tfrn / index.php

1  <?php
2  require_once('conf.php');
3  //exit;
4  ?>
5
6  <!DOCTYPE html>
7  <html lang="en">
8
9  <head>
10
11     <!-- Site inspiré de https://www.emse.fr/~pica...
12
13     <meta charset="utf-8">
14     <meta http-equiv="X-UA-Compatible" content="IE=edge">
15     <meta name="viewport" content="width=device-width, initial-scale=1">
16     <meta name="description" content="">
17     <title>Espace Emse - Espace Emse</title>
```

Commit Hash	Message	Time
TF	bug fix	avant-hier
TF	update website	il y a 3 jours
TF	nouvelle version site	2019-09-22
TF	html5 validation ok	2018-08-27
TF		2018-07-05

Dès que vous adoptez git, prenez les bons réflexes :

Créez un dépôt pour chacun de vos projets de code !

Quand vous souhaitez renommer/déplacer un fichier,
pour conserver toutes les infos de modification de code, utilisez :

git mv <path/nom_fichier> <nouveau_path/nom>

Quand vous avez une version fonctionnelle de votre code,
(en d'autres termes du code testé et qui marche !),
pensez à commiter et à pousser votre code !

Si votre nouveau code est non-opérationnel/nul, vous pouvez revenir
à une ancienne version commitée (**git reset ...**)

S'il y a des fichiers que vous ne souhaitez pas suivre dans votre dépôt (car ils sont temporaires, ou trop volumineux, etc.), à la racine de votre dépôt, créez un fichier « .gitignore ».

Dedans mettez les paths que vous ne souhaitez pas suivre, comme par ex :

```
<nom_fichier.ext>  
files/*.pdf
```

Pensez à ajouter ce fichier à votre dépôt (**git add .gitignore**) puis à partir de ce moment là, les paths incluent dans la fichier seront exclus du suivi.

Git est utile et performant sur du développement à plusieurs développeurs. Il faut alors utiliser plusieurs « branches ».

Dans la mesure du possible, évitez de travailler sur les mêmes fichiers, pour évitez les **merge** compliqués...

Faites toujours des commentaires compréhensibles (pour vous et pour les autres) dans vos commit.

Si jamais vous souhaitez renommer un commentaire « foireux » :

git commit --amend -m "Le nouveau commentaire."

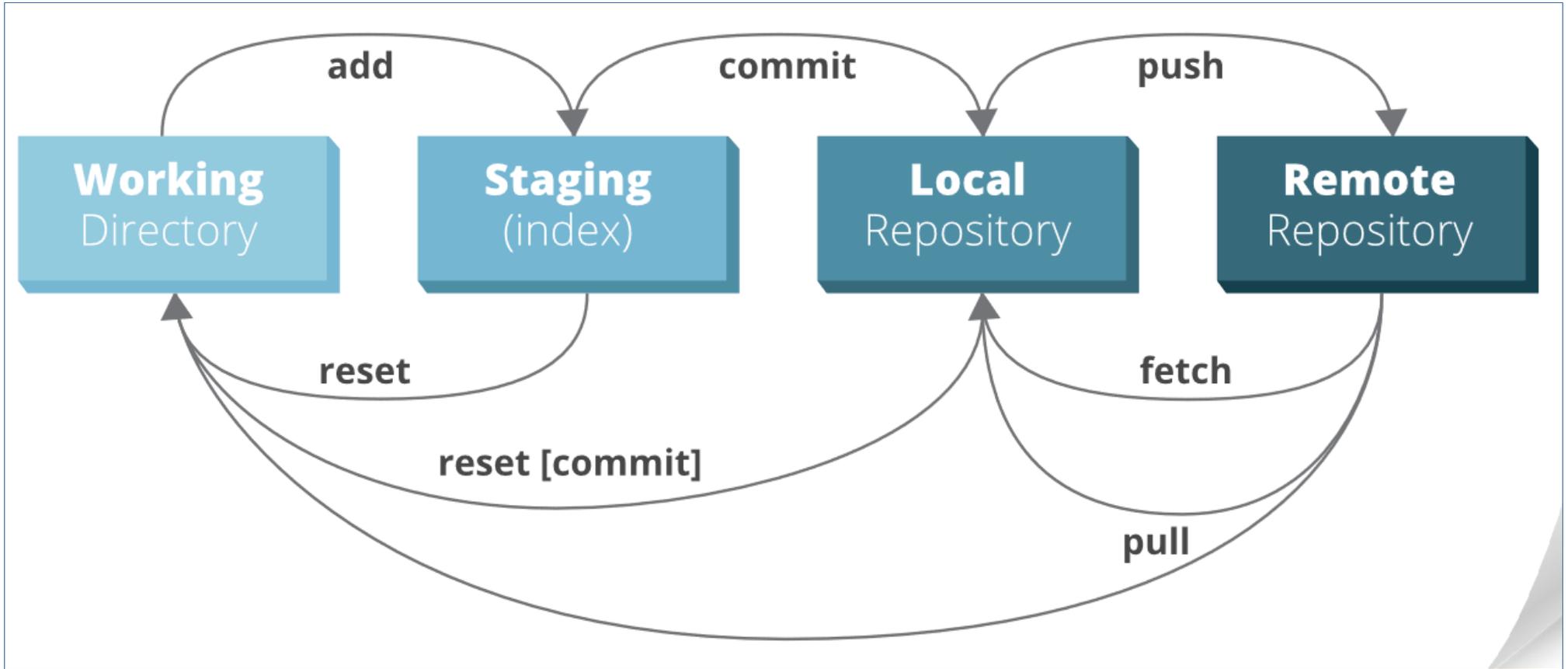
Uniquement si le commentaire n'a pas encore été diffusé, sinon voir :

<https://help.github.com/en/articles/changing-a-commit-message>

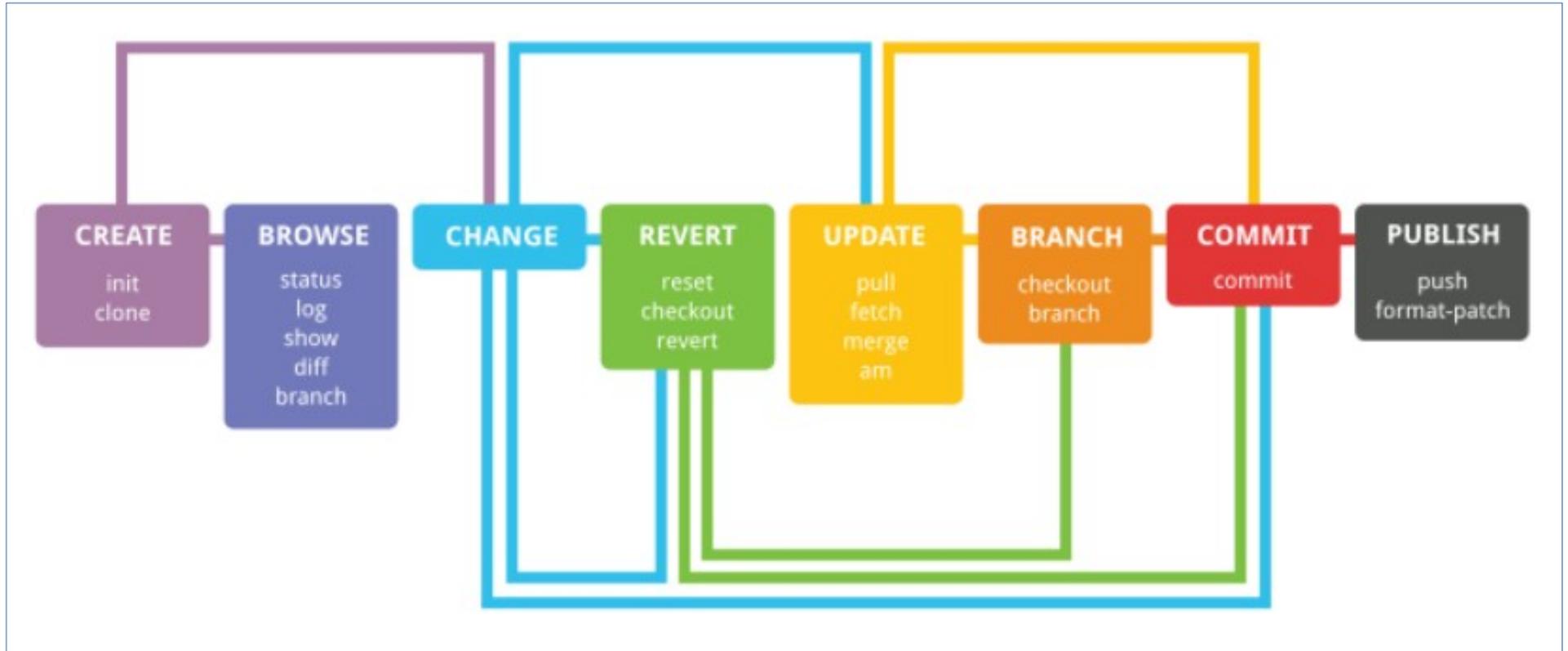
Pour voir le nb de commits / utilisateurs : **git shortlog -sn**

→ permet à un chef de projet de voir ceux qui ne font pas grand-chose...

Workflow



Workflow



Autres commandes utiles...

Antidater un commit :

```
GIT_AUTHOR_DATE="2009-11-26 08:32 +100" git commit -m "Un  
commit antidaté."
```

Abandonner le suivi de fichiers

```
git update-index --assume-unchanged README.md // début  
git update-index --no-assume-unchanged README.md // fin
```

Supprimer les fichiers inconnus de Git

```
git clean -n -d <path>
```

Exporter son projet dans une archive ZIP ou TAR.GZ

```
git archive --format=zip --prefix=mon_projet-1.0/ -9 1.0 >  
/chemin/de/destination/mon-projet-1.0.zip
```

Voir

<https://mirrors.edge.kernel.org/pub/software/scm/git/docs/git-archive.html>



Git Cheat Sheets

Allez sur votre moteur de recherche favori et adoptez votre git cheat sheet...

Celle que j'utilise personnellement :
http://files.zereturnaround.com/pdf/zt_git_cheat_sheet.pdf

Pour finir, bon à savoir :

Il existe des interfaces de bureau tierces pour consulter son code, les historiques, les branches, etc.

Interfaces de bureau tierces [\[modifier | modifier le code \]](#)

Il existe également d'autres applications offrant une interface de bureau :

- **qgit** propose des outils supplémentaires par rapport à **gitk** ;
- **Giggle** est une interface en **GTK+** ;
- **GitExtensions** est un client Git graphique tournant sous Windows ;
- **JGit/EGit** est une interface écrite en **Java**, puis une extension Git pour **Eclipse** ;
- **SourceTree** est un logiciel gratuit compatible Windows et OS X édité par **Atlassian** permettant de gérer des dépôts Git et **Mercurial** (commit, merge, fetch, push...) ;
- **Sublime Merge** : permet de lire un dépôt git afin d'afficher le différentiel des dernières modifications, la liste des soumissions (commits) et les branches, parmi d'autres choses (septembre 2018) ;
- **tig** est une interface texte libre (GPLv2), écrite en C ;
- **TortoiseGit** logiciel libre sous licence GPLv2, écrit en C/C++, reprend les éléments d'interface de **TortoiseSVN** ;
- **Tower** est un logiciel payant, uniquement pour OS X permettant la gestion de dépôts Git

Partage du dépôt avec l'enseignant !

Pour l'évaluation continue, partagez votre dépôt à Thomas.fressin@u-pem.fr :

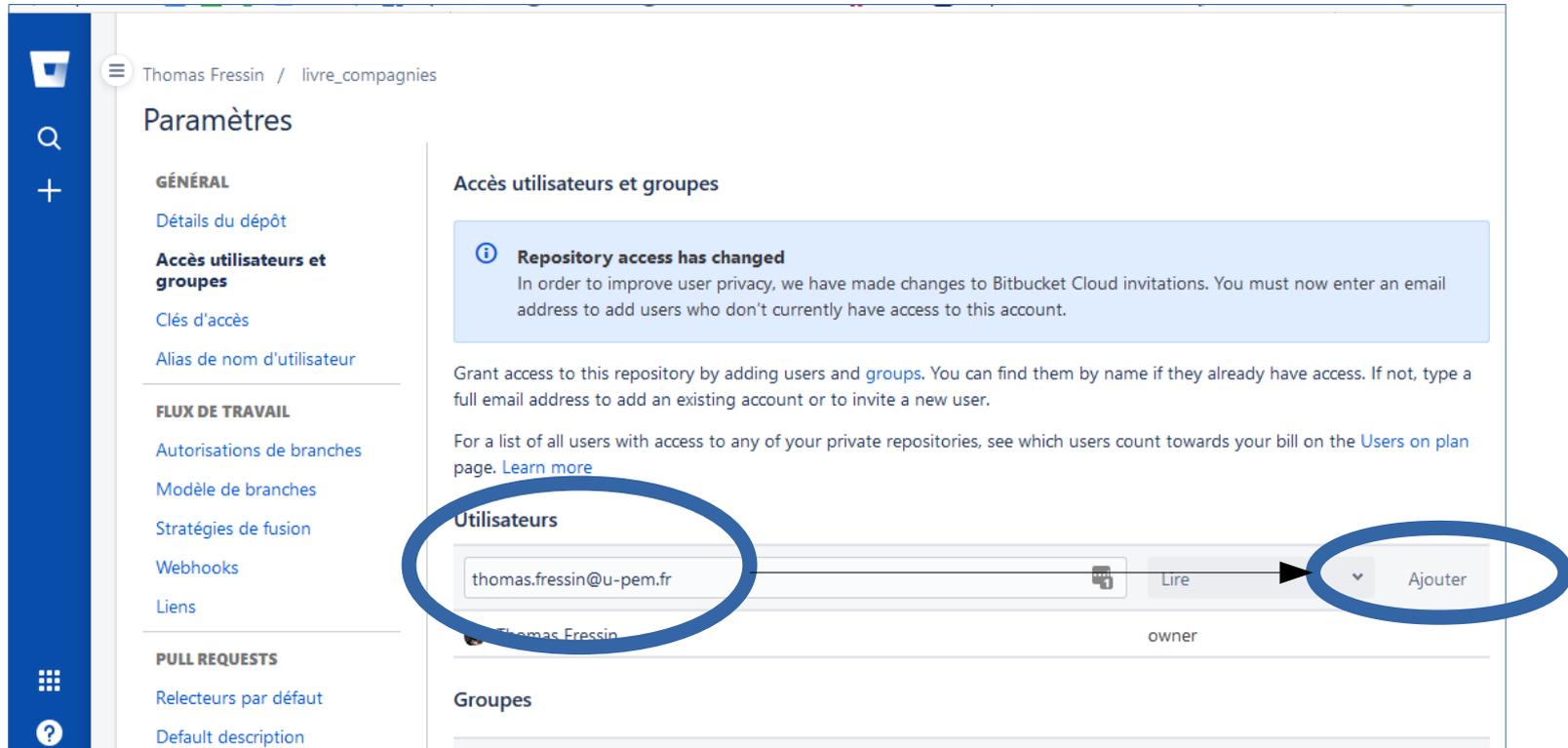


The screenshot shows the GitHub interface for a repository named 'livre_compagnies' by Thomas Fressin. The repository is currently on the 'master' branch. A dropdown menu is open, showing options: 'Closer', 'Add file', 'Manage notifications', 'Download repository', and 'Partager le dépôt'. The 'Partager le dépôt' option is circled in blue. The repository description states: 'C'est ici que vous trouverez les fichiers sources du dépôt. Si vous souhaitez donner aux utilisateurs une idée de ce qu'ils y trouvent, ajoutez une description à votre dépôt.' Below the description, there is a search bar for files and a table of repository contents. The table has columns for 'Nom', 'Taille', 'Last commit', and 'Message'. One folder named 'articles' is listed.

Nom	Taille	Last commit	Message
articles			

Partage du dépôt avec l'enseignant !

Pour l'évaluation continue, partagez votre dépôt à Thomas.fressin@u-pem.fr :



The screenshot shows the Bitbucket Cloud interface for a repository named 'livre_compagnies' owned by 'Thomas Fressin'. The 'Paramètres' (Settings) page is open, specifically the 'Accès utilisateurs et groupes' (User and group access) section. A blue notification box at the top states: 'Repository access has changed. In order to improve user privacy, we have made changes to Bitbucket Cloud invitations. You must now enter an email address to add users who don't currently have access to this account.'

Below the notification, there is a section titled 'Utilisateurs' (Users). A text input field contains the email address 'thomas.fressin@u-pem.fr'. To the right of the input field is a dropdown menu with a downward arrow and the text 'Ajouter'. A blue circle highlights the input field, and another blue circle highlights the 'Ajouter' button. An arrow points from the input field to the 'Ajouter' button. Below the input field, the user 'Thomas Fressin' is listed with the role 'owner'.

The left sidebar contains navigation options: 'GÉNÉRAL' (General) with 'Détails du dépôt', 'Accès utilisateurs et groupes', 'Clés d'accès', and 'Alias de nom d'utilisateur'; 'FLUX DE TRAVAIL' (Workflow) with 'Autorisations de branches', 'Modèle de branches', 'Stratégies de fusion', 'Webhooks', and 'Liens'; and 'PULL REQUESTS' with 'Relecteurs par défaut' and 'Default description'.

Règles importantes en développement :

1. Commitez votre code dès que vous franchissez une étape fonctionnelle significative, relative à une tâche que vous vous étiez fixé.
2. Commitez votre code avec des commentaires compréhensibles dans 3 ans.
3. Pensez à commiter/pousser votre code en fin de séance
4. N'hésitez pas à aller voir les versions précédentes de votre code si besoin.