

Marquer comme terminé

author: - François Rioult lang: fr

title: Installation de **React** sur la VDI (bureau distant)

Cette installation a été testée sur la VDI Ubuntu 22.04 (ou *bureau distant*), accessible par le client VMWare :

- soit le client lourd, installable sur tout type de machine. Sous Linux et les machines de l'université, dans un terminal taper `vmware-view` ou dans les applications, chercher **VMware Horizon Client**
- soit sur le client web, disponible à <https://bureau-distant.unicaen.fr>. Si vous n'êtes pas sur le réseau filaire de l'université, par exemple vous êtes en wifi sur **eduroam** ou depuis chez vous, vous devrez utiliser [la double authentification](#) pour obtenir le client web.

Les particularités du système de fichier

D'après [cette ressource](#).

Pour la VDI

Par le bureau distant, vous ne pouvez accéder qu'à votre dossier **Documents**, disponible également sur votre compte Linux ou Windows. Vous y avez un quota de 30GB. Ce stockage utilise le protocole SMB, qui interdit les liens symboliques.

Lorsque vous fermez la session de la VDI, le contenu de votre répertoire HOME est effacé. Il ne faut donc rien y stocker, et les paramètres des logiciels (entre autres l'historique et les identités gérés par le navigateur, mais également le fichier `~/.bashrc` seront systématiquement effacés).

Pour le compte Linux

Sur les machines de TP, votre répertoire HOME n'est pas effacé, il fait même l'objet de sauvegarde.

Attention : vous y avez un quota de 2GB. Soyez attentifs, cela arrive vite de saturer cet espace. En cas de problème de surcharge, utilisez l'utilitaire **ncdu** qui est correctement paramétré dans la session de notification qui s'affiche à l'ouverture de la session (utiliser le bouton **analyser avec ncdu**). Si vous avez fermé cette notification, vous pouvez l'obtenir à nouveau à l'aide de la commande `session_notify.ncdu` est indispensable pour visualiser les répertoires qui prennent de la place, appuyer sur **d** pour détruire, **q** pour quitter.

Commandes utiles

```
# déterminer le quota de ~ (limité à 2GB)
quota
# utilitaire de visualisation/gestion d'espace disque
ncdu
# déterminer le volume disque
du -hsx
```

Initialisation d'un projet **React Native**

- télécharger l'[archive du projet avec le dossier node_modules dont les liens symboliques sont déferencés](#)
- la désarchiver dans `~/Documents`
- renommer le dossier selon votre choix

Démarrage du projet

Une fois installé, il suffit de faire `npm run start` ou `npx expo`. Un serveur **expo** est lancé, vous permettant d'ouvrir l'émulateur web\ :

- appuyer sur **w**
- le projet est alors compilé avec **webpack**
- le navigateur est ouvert pour pointer sur <http://localhost:19006/>
- en cas de modification d'un fichier source, le projet est recompilé

En cas d'erreur de compilation, un message est affiché dans la console du compilateur, c'est-à-dire dans le terminal qui a lancé la compilation.

Exécution sur tablette/mobile

Votre tablette/mobile doit être sur le même réseau que la machine qui fait tourner le serveur **expo**. Si votre serveur est sur la VDI ou une machine de TP, ce n'est pas possible car le wifi **eduroam** est un réseau indépendant de l'infrastructure de l'université.

Vous devez donc faire tourner le serveur sur votre machine personnelle.

Sur votre tablette/mobile, installez l'application **Expo go** et fournissez lui l'adresse du serveur.

Vous pouvez également utiliser les serveurs d'**expo** pour construire un **APK Android**, installable sur votre device.

Gestion de la console de développement

Les erreurs de compilation/paquetage sont affichées dans le terminal qui a lancé la commande **npx expo** ou **npm run start**.

Si vous utilisez le navigateur pour tester votre projet, la console qui recoît vos affichages **console.log(...)** est celle du navigateur.

Si vous testez sur tablette/mobile, la console d'affichage est le terminal qui a lancé la commande **npx expo** ou **npm run start**.

Construction d'un projet sans liens symboliques

Il ne suffit pas d'utiliser **npm install --no-bin-links** pour utiliser un projet sans liens symboliques mais fonctionnel. En effet, le bundler **metro** est par exemple normalement appelé par un lien symbolique dans **node_modules/.bin/metro**, qui pointe vers **../metro/src/cli.js**.

Il faut donc retirer les liens en procédant comme suit\ :

- installer les modules de façon traditionnelle **npm install**
- utiliser **tar** avec l'option **-h** (ou **--dereference**) pour constituer une archive\ : les liens symboliques sont remplacés dans l'archive par le fichier vers lequel ils pointent
- désarchiver le dossier

La liste des commandes\ :

```
npm install
tar --dereference --create --file node_modules.tar node_modules
rm -fr node_modules
tar xf node_modules.tar
rm node_modules.tar
```

Modifié le: jeudi 31 août 2023, 18:37

◀ [Cours / TP - complet - en .pdf](#)

Choisir un élément

Aller à...

[Projet React Native vide ▶](#)

[mentions légales](#) . [vie privée](#) . [charte utilisation](#) . [unicaen](#) . [cemu](#) . [moodle](#)

[f](#) [t](#) [v](#) [@](#) [in](#)