António Domingos Ana Sofia

**Cahier de bord**

Argentin Yvann

Hottinger Jeremy

Lopes Marques Vasco

Table des matières

[Etape 1 3](#_Toc399872045)

[Description détaillé 3](#_Toc399872046)

[Semaine 1 3](#_Toc399872047)

[Rôles : 3](#_Toc399872048)

[Recherches : 3](#_Toc399872049)

[Codage : 3](#_Toc399872050)

[Résultats : 4](#_Toc399872051)

Création d’un mini OS simple en plusieurs étapes et par groupes.

# Etape 1

## Description détaillé

Implémenter un bootloader qui chargera un mini noyau à l’aide de l’émulateur QEMU.

* Créer une **image disque** qui servira de disque de boot à QEMU. Elle contiendra le secteur de boot suivi du mini kernel.
* Créer **2 fichiers en assembleur**, un pour le **bootloader** et un autre pour le **kernel**. Ils seront **compilés à l’aide d’un makefile**.
* Le **bootloader** devra **afficher** un message *« Loading kernel at address xxx... »* et **implémenter la lecture du kernel** une adresse choisie.
* Le **kernel** devra **afficher un message** permettant de vérifier qu’il a été correctement chargé.

## Semaine 1

### Rôles :

* Chef de projet : Hottinger Jeremy
* Rédacteur : António Domingos Ana Sofia
* Présentateur : Argentin Yvann
* Expert : Lopes Marques Vasco

### Recherches :

Vu que nous ne savions pas comment appliquer la théorie vue en cours, nous avons fait quelques recherche et avons trouvé un tutoriel[[1]](#footnote-1) qui explique bien les choses et nous nous sommes basé sur lui pour commencer cette étape.

### Codage :

|  |
| --- |
| *Insertion du code ici* |

### Résultats :

*Insertion print screen exécution code ici*

1. <http://viralpatel.net/taj/tutorial/hello_world_bootloader.php> [↑](#footnote-ref-1)