



BUT GAMES

---

# Rapport d'étude & notice d'installation

---

Installation de poste dans le cadre  
d'une nouvelle équipe de  
développeurs au seins de  
l'entreprise

---

Angelo CHAUVIN & Lorenzo AZZOPARDI

---



- 1. Préambule.....4
- 2. Dossier d'étude et choix des solutions.....5
  - 2.1 Choix de la distribution.....5
  - 2.2 Choix logiciels.....6
  - 2.3 Choix matériel.....6
- 3. Architecture logicielle.....7
- 4. Installation et configuration de la machine.....8
  - 4.1 Création de la VM.....8
  - 4.2 Initialisation de la machine.....9
  - 4.3 Configuration de la machine.....10
    - 4.3.1 Ajout d'alice en administratrice.....10
    - 4.3.2 Création de bob.....10
    - 4.3.3 Installation de Visual Studio Code.....10
    - 4.3.4 Installation de GIT.....10
    - 4.3.5 Changement du prompt shell.....10
    - 4.3.6 Génération de clé SSH.....11
    - 4.3.7 Installation de forth.....11
- 5. Notice d'utilisation.....11
  - 5.1 Vérifier l'installation de forth.....11
  - 5.2 Créer un projet forth.....11
  - 5.3 Portfolio.....12

## 1. Préambule

BUTGAMES est une entreprise de création de jeux. Dans le cadre du projet de réalisation d'un tout nouveaux jeu révolutionnaire multi plateforme écrit en *forth*, nous avons besoin de préparer des machines pour une nouvelle équipe de développeurs.

Pour faciliter la diffusion de ce nouvel environnement de développement, nous avons créé une machine virtuelle avec VMWare. Les détails des composants choisis lors de la création de cette machine seront transmis dans le rapport afin d'être diffusés auprès de l'équipe technique chargée d'installer les machines physiques.

Le travail a été effectué en deux semaines avec une grande liberté de choix de notre part car nous avons reçu un cahier des charges léger. Celui ci comportait les contraintes suivantes:

- Deux utilisateurs: Alice (administratrice) et Bob (développeur).
- Un projet *forth* doit être exécutable.
- VSCode doit être installé avec l'extension *forth*.
- La machine doit être en anglais avec un clavier français.
- git doit être installé.
- Le prompt bash doit afficher l'état du dépôt git du répertoire courant.
- Environnement léger basé sur linux.

Le travail a été effectué par l'équipe d'Administrateurs Système assurée par Angelo CHAUVIN et Lorenzo AZZOPARDI.

Ce rapport s'adresse au jury lors de la soutenance du 08/11/2024 ainsi qu'à la personne chargée de l'installation de la machine lors de cette même soutenance.

## 2. Dossier d'étude et choix des solutions

### 2.1 Choix de la distribution

Pour sélectionner l'environnement idéal, nous avons établi une liste de systèmes d'exploitation basés sur Linux et décidé de réaliser une analyse comparative. Cette analyse présente les avantages et inconvénients de chaque distribution, permettant ainsi de déterminer celle qui s'adapte le mieux aux exigences de notre projet. Le tableau suivant synthétise cette évaluation.

Critères	Puppy Linux	Xubuntu	Bodhi Linux	Lubuntu	Debian
Très léger	X				X
Démarrage rapide	X	X	X	X	X
Exécution RAM	X				
Basé sur Ubuntu		X	X	X	
Interface simple		X		X	X
Personnalisable		X	X		X
Pour ancienne machine	X	X	X	X	X
Large support		X		X	X
Interface rudimentaire	X				
Plus lourd		X			
Moins de fonctions	X		X	X	
Mauvaise prise en main	X		X		
Documentation limité	X		X		
Peu personnalisable				X	
Moins stable			X		

## 2.2 Choix logiciels

Pour ce projet, bien que les logiciels principaux aient été imposés, chaque composant de l'environnement a été sélectionné pour optimiser les performances et assurer la compatibilité. Voici un résumé des logiciels utilisés :

1. **Système d'exploitation** : Nous avons réalisé une analyse comparative de plusieurs distributions Linux légères, afin de choisir celle qui convient le mieux à notre projet. Cette sélection permet de maximiser les performances, même sur des configurations matérielles modestes.
2. **Visual Studio Code** : Éditeur de code imposé pour ce projet, VS Code est reconnu pour sa large gamme d'extensions, son intégration fluide avec Git, et sa compatibilité avec Linux, ce qui en fait un choix performant et populaire pour le développement.
3. **Git** : Le choix de Git a été imposé comme système de gestion de version. Sa popularité et ses fonctionnalités robustes en font un outil incontournable pour le suivi et la collaboration en développement.
4. **Forth** : Le langage Forth, également imposé, est utilisé pour les opérations de bas niveau, en offrant un contrôle précis sur les manipulations nécessaires.
5. **Script du prompt bash** : Nous avons décidé de créer notre propre script sh pour récupérer le status git du répertoire courant.

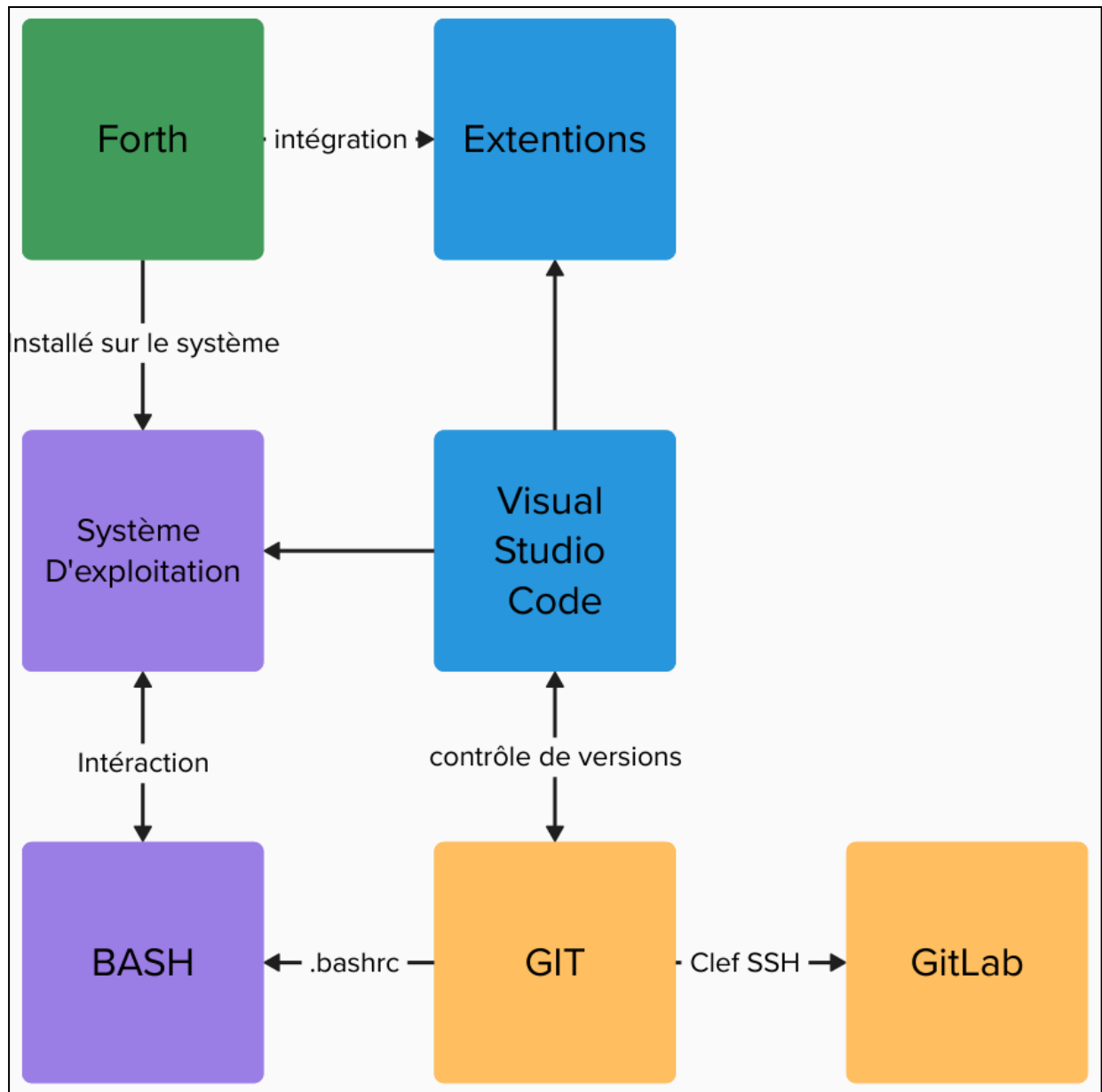
## 2.3 Choix matériel

Nous avons décidé de créer des machines ayant 8 Gio de RAM pour offrir un bon environnement de développement au développeur. 4 Gio de RAM était la quantité de RAM conseillée et 2 Gio était la quantité minimale conseillée. En offrant 8 Gio, on offre 6 Gio de RAM de plus que la quantité minimale ce qui permet au développeur de faire une grande quantité de tests avant d'être contraint à redémarrer sa machine. Il pourrait même ne pas avoir à la redémarrer.

Nous avons également décidé d'utiliser des processeurs à 4 cœurs car ces machines sont destinée à la création d'un gros jeu multiplateforme, donc, il est préférable d'avoir plusieurs cœurs pour les tests d'optimisation du code en faisant du *multiprocess* (processus consistant à répartir un code sur plusieurs code afin d'effectuer plusieurs tâches longues en même temps).

## 3. Architecture logicielle

L'illustration ci-dessous présente l'architecture logicielle de l'environnement de développement configuré pour l'équipe de développeurs. Cette machine virtuelle sous Linux intègre les composants principaux imposés pour le projet : Visual Studio Code, Git, et un environnement pour le langage Forth.



## 4. Installation et configuration de la machine

### 4.1 Création de la VM

Pour procéder à l'installation, veuillez vous rendre sur le site internet de Debian, "<https://debian.org>", cliquez sur "Téléchargement", cela va installer `debian-12.7.0-amd64-netinst.iso`.

- Ouvrez l'application "VMware Workstation 17 Player".
- Cliquez sur "Create a New Virtual Machine".
- Cliquez sur "Installer disc image file (iso)" puis sur Browse.
- Sélectionnez "`debian-12.7.0-amd64-netinst.iso`" et cliquez sur "Next >".
- Sélectionnez "Linux" dans "Guest operating system" et "Debian 11.x 64-bit" dans version puis cliquez sur "Next >".
- Renommez en "Debian 12" et cliquez sur "Next >".
- Laissez la page "Specify Disk Capacity" telle quelle et cliquez sur "Next >".
- Cliquez sur "Customize Hardware...".
- Dans "Memory" mettez la valeur à 8192.
- Dans Processors mettez "Number of processor cores" à 4 puis cliquez sur "Close".
- Cliquez enfin sur "Finish".



## 4.2 Initialisation de la machine

Après avoir lancé la machine, appuyez sur Entrée pour procéder à l'installation graphique.

- Appuyez de nouveau sur Entrée pour sélectionner la langue English.
- Tout en bas de la liste, cliquez sur "other" puis appuyez sur Entrée.
- Cliquez sur "Europe" puis appuyez sur Entrée.
- Cliquez sur "France" puis appuyez sur Entrée.
- Appuyez sur Entrée pour continuer avec la locale "en\_US.UTF-8".
- Cliquez sur "French" puis appuyez sur Entrée pour configurer le clavier en français.
- Après l'installation des composants, dans "Hostname", nommez la machine "but".
- Appuyez sur Entrée.
- Laissez vide "Configure the network" et appuyez sur Entrée.
- Dans "Root password" et dans "Re-enter password to verify" mettez "root@BUT".
- Cliquez sur "Continue".
- Mettez "Alice" dans "Full name for the new user" puis appuyez sur Entrée.
- Mettez "alice" dans "Username for your account" puis appuyez sur Entrée.
- Dans "Choose a password for the new user" et dans "Re-enter password to verify" mettez "alice@BUT" puis appuyez sur Entrée.
- Après l'installation, appuyez quatre fois sur Entrée dans "Partition disks".
- Cliquez sur "Yes" puis appuyez sur Entrée dans "Install the base system".
- Cliquez sur "No" puis appuyez sur Entrée dans "Configure the package manager".
- Cliquez sur "France" puis appuyez sur Entrée.
- Laissez "deb.debian.org" puis appuyez sur Entrée.
- Laissez vide "HTTP proxy information" puis appuyez sur Entrée.
- Après l'installation, cliquez sur "no" puis appuyez sur Entrée.
- Dans "Software selection" vérifiez que les logiciels suivants soient bien sélectionnés:
  - Debian desktop environment
  - ... Xfce
  - SSH server
  - standard system utilities
- Ne sélectionnez surtout pas "... GNOME" puis appuyez sur Entrée.
- Après l'installation cliquez sur "Yes" et appuyez sur Entrée.
- Cliquez sur "/dev/sda" puis appuyez sur Entrée.
- Appuyez sur Entrée pour reboot.

## 4.3 Configuration de la machine

### 4.3.1 Ajout d'alice en administratrice

Lorsque la machine est allumée, connectez-vous en tant qu'Alice

- Ouvrir le Terminal.
- Taper "su root".
- Écrivez le mot de passe du root (root@BUT).
- Taper "sudo visudo".
- Rajouter la ligne "alice ALL=(ALL:ALL) ALL".
- Faites CTRL + o, Entrée, CTRL + x.
- Taper "exit".

### 4.3.2 Création de bob

Lorsque la machine est allumée, connectez-vous en tant qu'Alice

- Tapez "sudo adduser".
- Tapez le mot de passe d'Alice (alice@BUT) si demandé.
- Tapez "bob@BUT" pour "New password" et "Retype password".
- Tapez "Bob" pour "Full Name".
- Ignorez "Room Number", "Work Phone", "Home Phone" et "Other".

### 4.3.3 Installation de Visual Studio Code

Pour installer Visual Studio Code, lancer le navigateur internet de la machine virtuelle.

- Rendez-vous sur "<https://code.visualstudio.com/download>".
- Cliquez sur ".deb".
- Retournez dans le Terminal.
- Tapez "cd Downloads".
- Tapez "ls" et notez le nom du fichier téléchargé.
- Tapez "sudo dpkg -i <nom\_du\_fichier\_téléchargé>".
- Tapez le mot de passe d'Alice (alice@BUT) si demandé.
- Appuyez sur Entrée lorsque la pop up s'affiche.

### 4.3.4 Installation de GIT

Installons GIT en tapant dans Terminal "sudo apt install git -y"

### 4.3.5 Changement du prompt shell

Connectez vous en tant qu'Alice pour effectuer la manipulation et ouvrez un terminal.

- Tapez "sudo wget https://github.com/pazzazzo/butgames/raw/refs/heads/main/git.sh -O /home/bob/git.sh"
- Tapez "sudo chmod +x /home/bob/git.sh"
- Tapez "sudo echo ".git.sh" | sudo tee -a /home/bob/.bashrc"

# INSTALLATION DES POSTES

## 4.3.6 Génération de clé SSH

Pour cloner un dépôt de l'IUT, vous devez avoir une clef SSH, pour cela, connectez vous en tant que bob et ouvrez un terminal

- Tapez "ssh-keygen -t rsa -b 4096 -C "bob@but".
- Appuyez sur Entrée jusqu'à ce que la clé soit générée.
- Tapez "cat ~/.ssh/id\_rsa.pub" et copiez le résultat
- Connectez vous sur <https://gitlab-ce.iut.u-bordeaux.fr/>
- Cliquez sur votre photo de profil puis sur "SSH Keys".
- Collez la clé dans le champ Key.
- Donnez un titre et appuyez sur "Add key".

## 4.3.7 Installation de forth

Connectez vous en tant qu'Alice puis ouvrez le terminal

- Tapez "sudo apt install gforth dos2unix"

## 5. Notice d'utilisation

### 5.1 Vérifier l'installation de forth

Après avoir installé la machine, il est nécessaire de vérifier le bon fonctionnement de tous les composants, pour cela, un repo est accessible sur le gitlab de l'IUT.

Afin de faire ce test, veuillez suivre les étapes suivantes:

- Connectez vous en tant que bob.
- Ouvrez un terminal.
- Placez vous dans le dossier qui accueillera le dépôt avec la commande "cd".
- Tapez "git clone [git@gitlab-ce.iut.u-bordeaux.fr:rgiot/hello.sae4.but1.iut.git](https://gitlab-ce.iut.u-bordeaux.fr:rgiot/hello.sae4.but1.iut.git)".
- Tapez "cd hello.sae4.but1.iut".
- Vous pouvez taper "git status" pour voir le status du dépôt.
- Vous pouvez taper "ls -a" pour voir le contenu du répertoire
- Tapez "nano ./test.sh"
- Supprimez la seconde ligne
- Faites Ctrl + O, Entrée, puis Ctrl + X
- Tapez "dos2unix ./test.sh"
- Tapez "./test.sh"

Si tout a été bien installé, l'exécution du script devrait afficher: "Hello World !"

### 5.2 Créer un projet forth

Dans le terminal, allez dans le répertoire dans lequel initialisez le projet puis tapez la commande: "code ." afin d'ouvrir VSCode dans le répertoire.

Créez un fichier "main.fth" puis écrivez votre code.

Pour exécuter votre programme, tapez dans le terminal "gforth main.fth"

## 5.3 Portfolio

Dans ce projet, nous avons configuré un environnement Linux optimisé pour une équipe de développement Forth, avec Git et Visual Studio Code, tout en gérant des permissions spécifiques pour chaque utilisateur. Une difficulté notable a été la création de l'utilisateur bob, car l'utilisation de useradd au lieu de adduser a initialement causé des problèmes de configuration d'accès. Nous avons également personnalisé le prompt Bash pour afficher l'état Git, facilitant la gestion du code. Ces ajustements offrent un environnement de travail sécurisé et fonctionnel pour les développeurs Forth.

Voici la capture d'écran de VSCode sur la machine:

