16/12/2016

**Projet n°2 : eXiaSaver**

Projet n°2 Groupe 2

eXia A1.

# Table des matières

[Table des matières 1](#_Toc469575424)

[Contexte du projet 2](#_Toc469575425)

[Fonctionnement du programme 2](#_Toc469575426)

[Travail réalisé 4](#_Toc469575427)

[Lanceur 4](#_Toc469575428)

[screenSaver Statique 4](#_Toc469575429)

[screenSaver Dynamique 4](#_Toc469575430)

[screenSaver interactif 4](#_Toc469575431)

[Manuel d’utilisation du programme 5](#_Toc469575432)

[Conclusion du projet 5](#_Toc469575433)

[Bilan personnel du projet 6](#_Toc469575434)

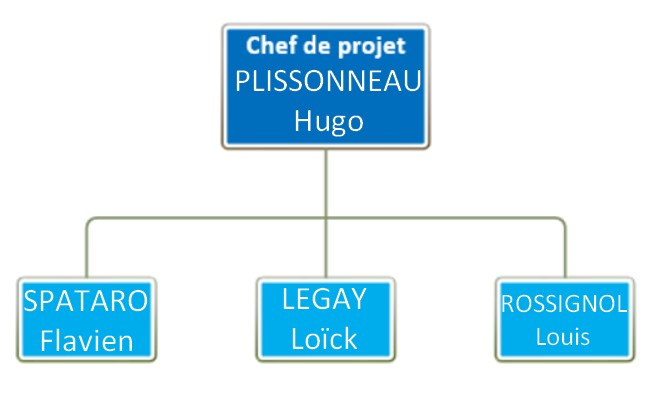
[ROSSIGNOL Louis : 6](#_Toc469575435)

[LEGAY Loïck : 6](#_Toc469575436)

[SPATARO Flavien : 6](#_Toc469575437)

[PLISSONNEAU Hugo : 6](#_Toc469575438)

# **Présentation de l’équipe**



Notre équipe est composé de 4 membres, 3 développeurs, qui sont Flavien ; Loïck et Louis, ainsi que d’un chef d’équipe, Hugo.

Notre groupe étant plutôt homogènes, nous nous aidons mutuellement afin d’avancer rapidement dans notre projet.

# **Contexte du projet**

Nous devions réaliser un gestionnaire de screenSaver appelé « exiaSaver ».

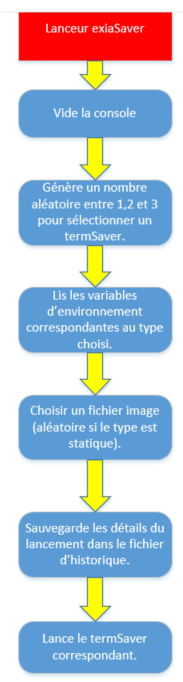
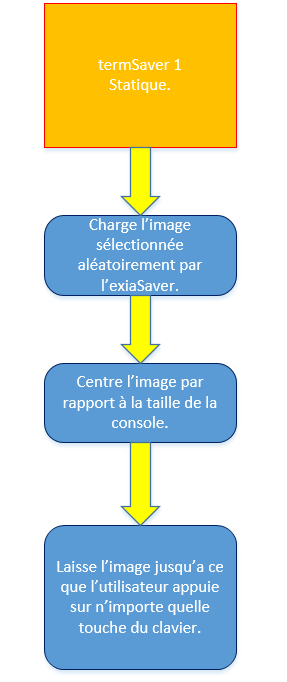
Le but était de créer un lanceur choisissant aléatoirement entre 3 screenSaver différents :

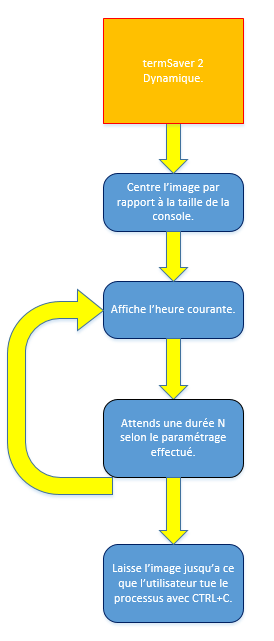
* **Un screenSaver statique** : c’est-à-dire une image simple, qui s’effacera lorsque l’utilisateur pressera une touche de son clavier.
* **Un screenSaver dynamique** : qui affichera l’heure actuelle avec une actualisation, et cette fois l’utilisateur devra « tuer » le processus avec la commande ctrl + C afin de terminer le programme.
* **Un screenSaver interactif** : ou l’utilisateur devra indiquer des commandes de déplacement à un avion, qui se déplacera selon ces dernières.

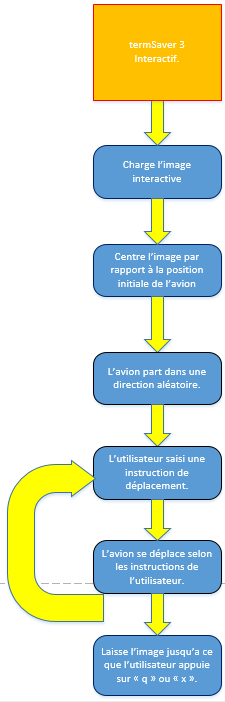
# **Fonctionnement du programme**

Afin de simplifier l’explication des programmes, nous avons choisi de reprendre nos diagrammes réalisé grâce à Visio pour la feuille d’avancement, car ils reprennent étape par étape le lanceur ainsi que les 3 termSaver.

Ils expliquent étapes par étapes ce que le programme va faire, tout le fonctionnement est indiqué ici.







# **Travail réalisé**

## Lanceur

Le lanceur est fonctionnel, il vide la console à son lancement, génère un nombre aléatoire afin de choisir l’un des screenSaver et en fonction de ce nombre exécute l’un des programmes, il n’arrive cependant pas à choisir l’interactif, car nous n’avons pas eu le temps de le réaliser.

Il permet aussi d’afficher les statistiques avec le paramètre -stat.

Il correspond donc à nos attentes, excepté pour le screenSaver interactif.

## screenSaver Statique

Lorsqu’il est exécuté par le lanceur, il choisit aléatoirement l’une des images .pbm et la charge dans la console, après l’avoir centré horizontalement et verticalement.

Ce ne sont que des images en noir et blanc, nous n’avons pas réussi à coder la partie concernant les couleurs, mais il est possible de sélectionner un paramètre pour choisir l’image que le screenSaver affichera. Le screenSaver se quitte automatiquement dès que l’on clique sur une touche.

## screenSaver Dynamique

Il arrive à afficher l’heure actuelle et à l’actualiser au bout d’un certain laps de temps (par défaut, il est de 10 secondes).

Il est aussi possible de le lancer avec un paramètre, pour modifier ce laps de temps.

Nous n’avons pas réussi à faire deux formats d’images différents pour ce screenSaver.

## screenSaver interactif

Ce screenSaver n’a pas été réalisé par manque de temps,

# **Manuel d’utilisation du programme**

Afin de lancer le programme, placez-vous dans le répertoire contenant le programme puis entrez la commande suivante : « ./exiasaver ».

Le programme se chargera lui-même de lancer l’un des 3 différents termSaver, grâce à un choix aléatoire, qui chargera les variables d’environnements nécessaires.

Le lancement de notre exiaSaver peut se faire avec des paramètres, en effet la commande « ./exiasaver » permet simplement d’exécuter le programme simplement, tandis que la commande « ./exiasaver -stat » affichera les logs d’utilisations ainsi que les statistiques de lancement.  
Le fonctionnement du programme est différent selon le paramètre entré.

L’exiaSaver n’est pas le seul moyen d’utiliser les différents termsaver, il existe un moyen pour lancer manuellement les termsaver sans passer par notre lanceur, pour cela il est nécessaire de se placer dans les différents dossiers, soit celui du statique ou du dynamique, grâce aux commandes de bases de linux. ( « cd LanceurStatique » ou « cd LanceurDynamique » ), puis d’utiliser la commande d’exécution suivie du nom du termsaver (« ./statique » ou « ./dynamique »).

Il est aussi possible de modifier certains paramètres du programme, pour le statique une image particulière, en entrant le numéro d’une image (Ces numéros allant de 1 à 5), par exemple « ./statique 3», et pour le dynamique il est possible de modifier le temps de rafraîchissement de l’heure, en tapant ceci : « ./dynamique X », où « X » est le laps de temps voulu.

Le plus simple étant d’exécuter le lanceur directement, étant donné qu’il gère automatiquement le choix du termSaver et des images.

# **Conclusion du projet**

De manière générale, ce projet nous pousse à développer nos connaissances en langage C, et à utiliser le plein potentiel d’une machine virtuelle sous linux, et surtout de gérer un groupe en situation d’entreprise, à réaliser un vrai travail d’équipe tout en respectant certaines contraintes.

Nous avons réalisé les termSaver statique et dynamique, et réussi à faire fonctionner notre lanceur pour qu’il exécute correctement ses fonctionnalités.

# **Bilan personnel du projet**

## ROSSIGNOL Louis :

Ce projet m’a permis de mettre en application les différents Prosits étudiés (multiprogrammation) en utilisant un nouvel système d'exploitation, tout en poussant mes recherches plus loin, grâce à ce projet j'ai compris le fonctionnement en mémoire des tableaux, string ainsi que les listes chainées. Ce projet m'a donné une bonne vision du travail en groupe car tous les membres de l’équipe étaient impliqués au même niveau.

## LEGAY Loïck :

Ce projet était très intéressant pour approfondir nos connaissances sur la gestion de mémoire en C, l'environnement linux et apprendre la gestion de lecture d'image pbm.

Le groupe était très investi pour arriver au résultat attendu, ce qui nous a permis d’avancer rapidement dans ce projet.

## SPATARO Flavien :

Le projet exiasaver était un bon exercice pour voir nos capacités de programmation en C et la gestion de Linux. L'équipe était très motivée ce qui a permis une bonne cohésion.

## PLISSONNEAU Hugo :

Étant donné que je ne connaissais pas énormément de chose en programmation, réaliser ce projet m’a permis d’approfondir tous ce que je savais des prosit, et à concrétiser tout le côté « texte » des exemples que nous avions, en réalisant différents programmes (Je pense notamment aux pointeurs).

En tant que chef de projet j’ai aussi pu expérimenté la gestion d’une équipe, de la répartition du travail à la motivation du groupe, afin d’avoir une bonne ambiance de travail et des résultats satisfaisant.