



## Introduction to Operating Systems (IOS) 2021-2022 Global evaluation - Project Specification

### DEVELOPMENT OF A TEXT ADVENTURE ESCAPE ROOM GAME

#### 1.- Introduction

This is a hiring process. This company needs people with the ability to produce new applications for the Linux operating system in C using its system call interface and the libraries (programming interface). In this case you are working alone. You will not have to manage as a team and therefore all other requirements will be discussed in greater depth.

In the following pages you will get all the information the company can give you to develop the project and all the evaluation criteria. You will have to manage with all other need you have. You can always consult the manager (Edurne) who will always be happy to answer to your requests, if possible.

#### 2.- Description of the project

You must develop a limited shell (a computer program which exposes an operating system's services to a human user or other program) that can execute the commands you have developed or any other Linux command.

##### 2.1.- Requirements of the project

###### 2.1.1.- The shell commands

###### Minimum commands you have to develop:

1. Read the contents of a directory (just the name or more information) – (ls -l)
2. Read from file or stdin and print in stdout – (cat)
3. Create new names for a file - (link)
4. Copy a file – (cp)

###### 2.1.2.- The code

- The project should include the use of the following system calls and library functions:
  - **Input/Output and file system management**
    - System calls
      - open
      - read
      - write
      - close
      - link
      - unlink
      - stat
      - readdir
      - lseek
  - C library functions
    - opendir
    - closedir
  - **Process management**
    - System calls
      - fork
      - wait (**not waitpid nor waitid**)
    - C library functions
      - execlp or execvp
- Concatenate commands with pipes

- Error control
  - All errors that could occur during execution should be handled and users should be informed about why this error occurred. Test in bash a command that may cause an error and see how the user is informed, for example:
    - `$ ls aÑsldk`  
`ls: cannot access aÑsldk: No such file or directory`
- Every command should have a man page with its specification.

### 2.1.3.- The project report

Each proposer must deliver before midnight of the **20th may** a PDF document containing:

- The full path to the project's directory on the server.
- The description/design of the full project.
- The verification document of each command demonstrating that all the possible cases have been tried.
- An explanation of where in the code can be found each of the worked theoretical concepts, such as:

#### **Project:**

3.- I/O and files API (mandatory)

+ Advanced access to device properties (optional)

4.- User management API (optional)

5.- Memory management API (optional)

6.- Process and thread management API (Process mandatory / Threads optional)

7.- Pipes management API (mandatory)

- A section with the proposer's conclusions.

### 2.3.- Evaluation headings (rubrics)

<b>Trait</b>	<b>Good</b> <sub>(1.5)</sub>	<b>Satisfactory</b> <sub>(0.7)</sub>	<b>Unsatisfactory</b> <sub>(0.2)</sub>	<b>Not tried</b> <sub>(0.0)</sub>
The shell works	The shell includes very creative uses	The shell works as expected	The shell doesn't work as expected	
Commands are independent and work as specified		All the commands that could be independent are and can be properly executed on their own	Some of the commands that could be independent are not independent or can not be properly executed on their own	
Use of required system calls and library functions	All the required system calls and library functions have been correctly used and other system calls have also been correctly used	All the required system calls and library functions have been correctly used	Not all the required system calls and library functions have been used or some of them are not correctly used	
Use of pipes to concatenate commands	Pipes have been used for other more creative uses	Pipes have been correctly used to concatenate commands	Pipes have not been used or have not been correctly used	
Error control	Every possible error has been controlled and reported on the verification documents	Most every possible error has been controlled and reported on the verification documents	Poor error control reported on the verification documents	
Man pages	The man pages are useful for the users that don't know how to use the commands and provide additional and very useful information	The man pages are useful for the users that don't know how to use the commands	The man pages, if available do not help the user on the use of the commands	
Report	The report surpasses the expectations. It has a very clear structure, is very well written and helps in every step to understand the project	The report is useful to find out the important parts of the code, to understand the development process or the operation of the game and is well written	The report is not useful to find out the important parts of the code, to understand the development process or the operation of the game, or is very poorly written	