

Turma 3 Grupo 1 Project Specification

WINDOW DISPLAY SERVER AND WINDOW MANAGER

We intend to make a barebones X window system clone with a window manager. Our driver like the X window system will consist in two parts, the compositor and the server. The compositor which is responsible of drawing the windows and the server which receives all the communications from the kernel and from the processes using the exposed API.

For the project we'll develop an API that exposes some functions to allow window creation and management. Inter process communication(IPC) will be key and since Minix already has library which eases this part we'll be using it. In cases where 56 bytes might not be enough, e.g during window creation, grants will play an important role.

The **graphics card** will be used to render the desktop environment and all the windows. The **mouse** will be our pointer, the first input method the user can use to interact with the system.

Then, the **keyboard** will be the second input method. By sending keystrokes the user can open/close windows and even interact with some menus!

UART might be using for debugging purposes only. E.g receiving all messages from the xtrace equivalent.

Serial port might be used by some apps we develop as examples or for a window system over the internet.

RTC can be used for the time and for a schedule example app.

After a basic API is exposed we'll start to develop in order to test/demonstrate its use cases. These apps will be essential in terms of showing the power of the window server and manager.