Universidad Nacional de Colombia - sede Bogotá
Facultad de Ingeniería
Departamento de Sistemas e Industrial
Curso: Ingeniería de Software 1 (2016701)

COMPILE C++ FILE	
ACTORS ● User ● Compilation Module	**REQUIREMENT** FR_12 – The system must compile C++ files directly from the editor.
DESCRIPTION This use case describes how a user compiles a C++ source file using the built-in functionality of the editor. The system uses a predefined C++ compiler and displays any compilation errors or success messages within the interface. The system has a built-in compiler setting, though they may be changed by the user	
PRE-CONDITIONS ● A C++ file must be open and saved. ● The file must have a valid name and extension (e.g., .cpp).	
NORMAL FLOW 1. The user clicks the “Compile” button or selects “Compile” from the menu. 2. The system identifies the active C++ file and its path. 3. The system invokes the compiler with the appropriate flags. 4. The system captures the output of the compilation process. 5. If successful, the system indicates compilation success.	
ALTERNATIVE FLOW 1. If the file is not saved: 1.1. The system saves it automatically and then compiles the code. 2. If an error occurs during compilation: 2.1. The system will stop the compilation process. 2.2. The error which caused the compilation to stop will be shown to the user.	
POST-CONDITIONS ● A compiled executable is generated (if successful). ● The user is informed about the status of the compilation.	
NOTES ● Compiler paths and flags should be configurable in system settings. ● The system should allow compilation from keyboard shortcuts as well.	