Zad 1.

roller\_shutter.cpp



roller\_shutter\_builder.h

roller\_shutter\_builder.cpp



**Zad. 2**

W naszym przypadku wzorzec “budowniczego” nieszczególnie się sprawdził, ponieważ nasze urządzenia nie maja wiele opcji konfiguracyjnych, bardziej przydatny może być wzorzec "fabryki" (factory pattern), który umożliwia tworzenie różnych rodzajów obiektów w zależności od potrzeb, ale nie jest skomplikowany tak jak wzorzec budowniczego.

**Zadanie Refaktoryzacja**

Nawet jeśli kod jest względnie czytelny, warto dodać komentarze, które wyjaśnią, co robią poszczególne fragmenty kodu. To może być pomocne dla innych, którzy czytają kod.

Podzielenie długich funkcji, na mniejsze funkcje, które wykonują konkretne zadania, aby kod byl bardziej zrozumiały.