# Detailed Design Patterns

The most challenging module of our architecture is "Search Documents".

### GRASP patterns

**#1 Creator:** The class "Search Adapter" closely uses the class "Search Interface".

**#2 Expert:** The class "Compose Adapter" has the information needed to assign the responsibility to instance of the class "Image Interface" and "XML Interface"

**#3 Low coupling:** The class "Compose Adapter" is used in order to minimise the coupling between "View Document(s)", "Image Interface" and "XML Interface"

**#5 High cohesion:** In order to maximise cohesion the class "Search Adapter" combines two main functionalities "simple search" and "advanced search".

**#6 Polymorphism:** The class "Search Adapter" handle alternatives based on applied search filters.

### GoF patterns

**#1 Adapter:** We resolve the incomplete interface problem for the view document scenario by introducing the class "Compose Adapter" which combines the result from the classes "Image Interface" and "XML Interface"

**#3 Singleton:** The class "Search Adapter" is constructed in with two functionalities "simple_search" and "advanced_search" that will adapt different types of filters to the search interface, requiring only one instance of the class itself.