

Assignment 1:KWIC-KWAC-KWOC

Code Repository URL:

<https://github.com/boyangwang/CS3213-assignment-1.git>

Name	Wang Boyang	Le Beier
Matriculation Number	A0078695H	A0097708N

1. Introduction

Key Word In Context(KWIC) index systems takes in a list of lines and circularly shifts the line by moving the first word to the back of the line. The system outputs the list in alphabetical order. Therefore this assignment uses KWIC as an example to demonstrate one of the system architecture mentioned in the article “An Introduction to Software Architecture” by David Garlan and Mary Shaw.

2. Design

Our architecture design follows the ADT design paradigm, as illustrated in Fig 1. Use interacts with KWIC UI, which accepts a command serial number and invokes the corresponding command handler class. These command handler classes will then process the input, invoke utility classes like FileInputHandler if needed, and return the output to KWIC UI to display. The underlying data structure and algorithm are encapsulated in EntryProcessor and EntryManager classes.

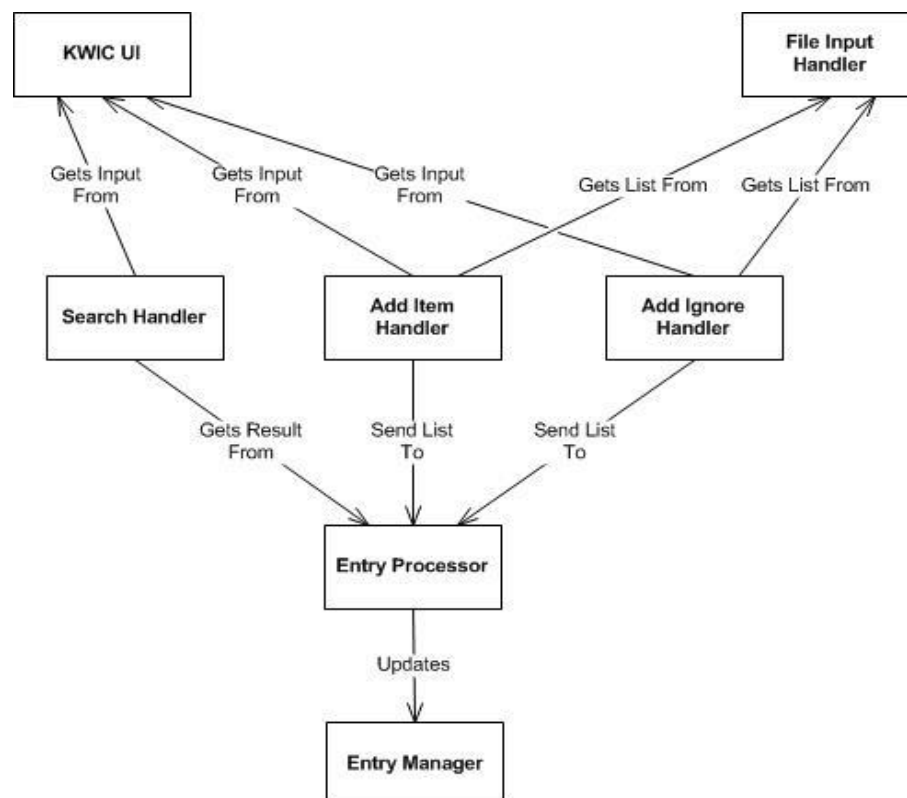


Fig 1. Software Architecture Diagram

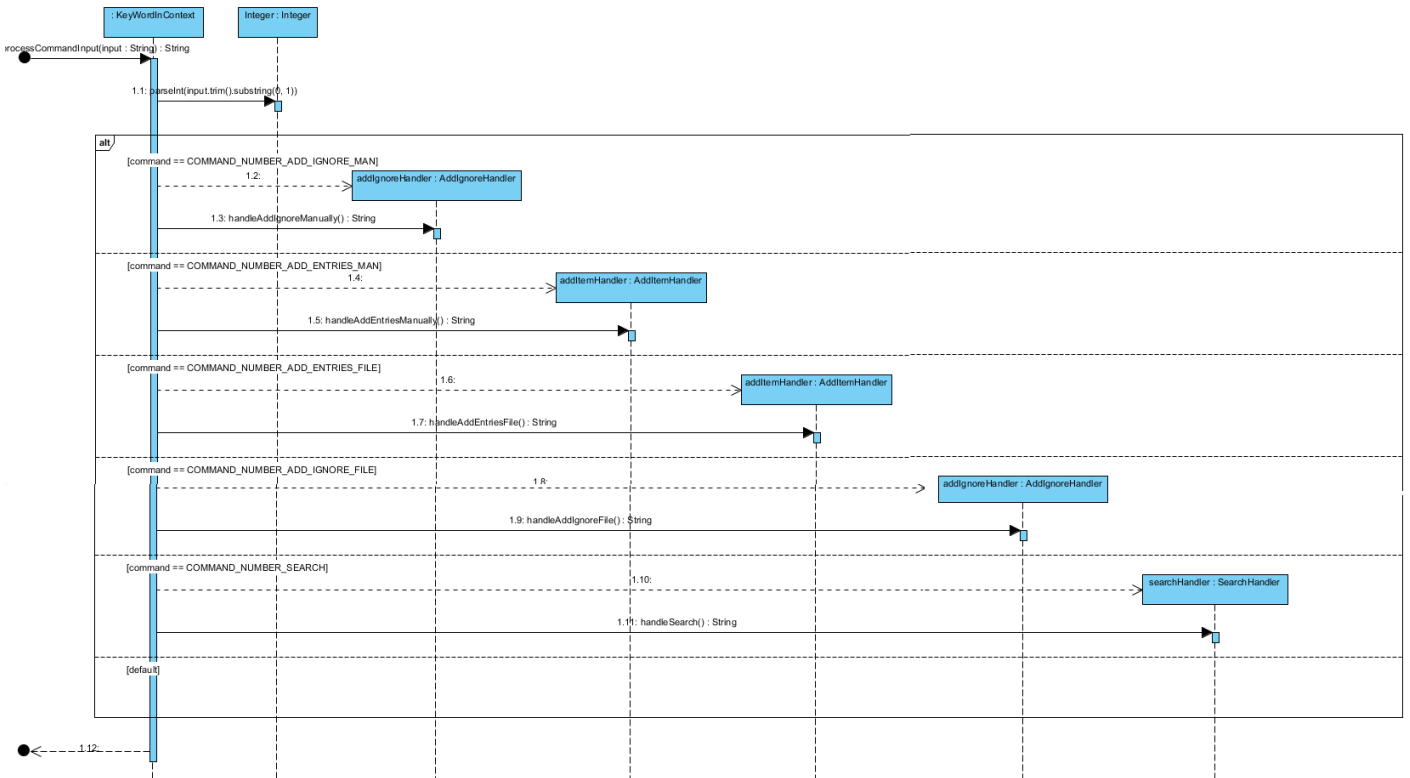


Fig 2. Sequence diagram of processCommandInput(String)

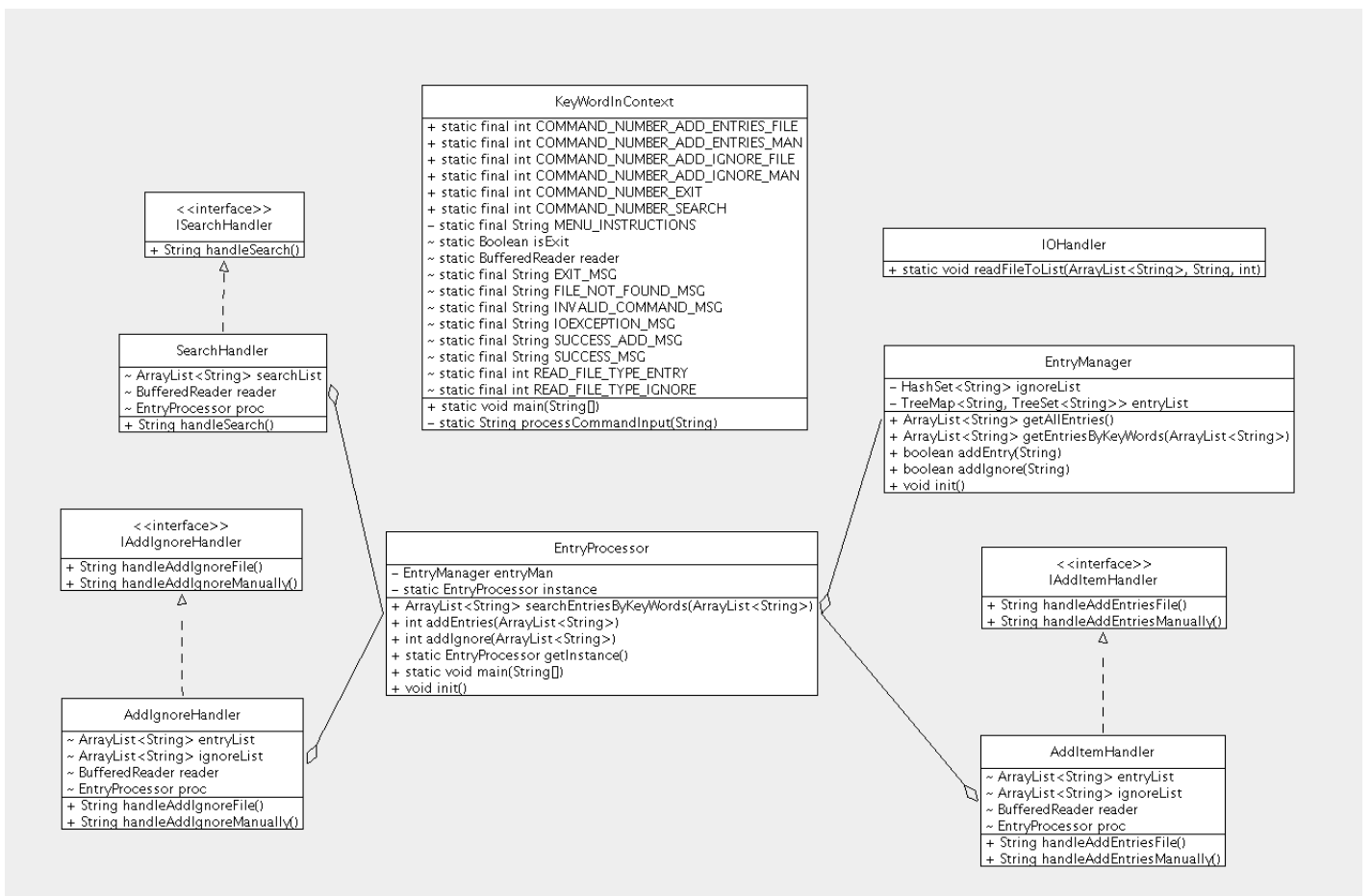


Fig 3. Class diagram of KWIC

3. Limitations and Benefits of Selected Design

Benefits	Limitations
<p>1.Low coupling between the command handlers and the Entry Manager class. This allows easy enhance on individual components without affecting the rest.</p> <p>2. Each class encapsulates exactly one functionality making the design highly coherent.</p> <p>3. API between command components are extracted out in interfaces which allows easy substitution.</p> <p>4. Allows easy extension of new commands. (The only class that needs to be modified is keyword in context class)</p>	<p>1. The use of singleton pattern in the design makes it difficult to implement interface for those singleton classes.</p> <p>2. Lack of UI layer which caused the UI to be coupled with the implementation.</p>