

Corso di Software Architecture Design

Presentazione dei risultati della Iteration Review n. 3

6 Giugno 2023

- **Identificativo del Team:** **G8**
- **Componenti del Team:**
Abdel Majid Zaira, Cipollaro Daiana, Di Serio Francesco, Manco Lorenzo
- **Identificativo del Task:** **T6**

Obiettivi dell'Iterazione

- O1: Implementazione dinamica del modello
- O2: Aggiunta di funzionalità dell'editor
- O3: Studio e approfondimento delle tecnologie per la generazione della coverage
- O4: Raffinamento della visualizzazione della classe da testare

Attività Svolte

Attività di ingegneria del software avviate o svolte

- Raffinamento dei requisiti
- Studio di tecnologie software
- Studio di Framework
- Sviluppo di codice
- Raffinamento dell'interfaccia
- Definizione delle interfacce
- Raffinamento dell'architettura

Altre attività organizzative/ di supporto al team svolte

- Riunioni di gruppo tramite MS Teams
- Riunioni in presenza

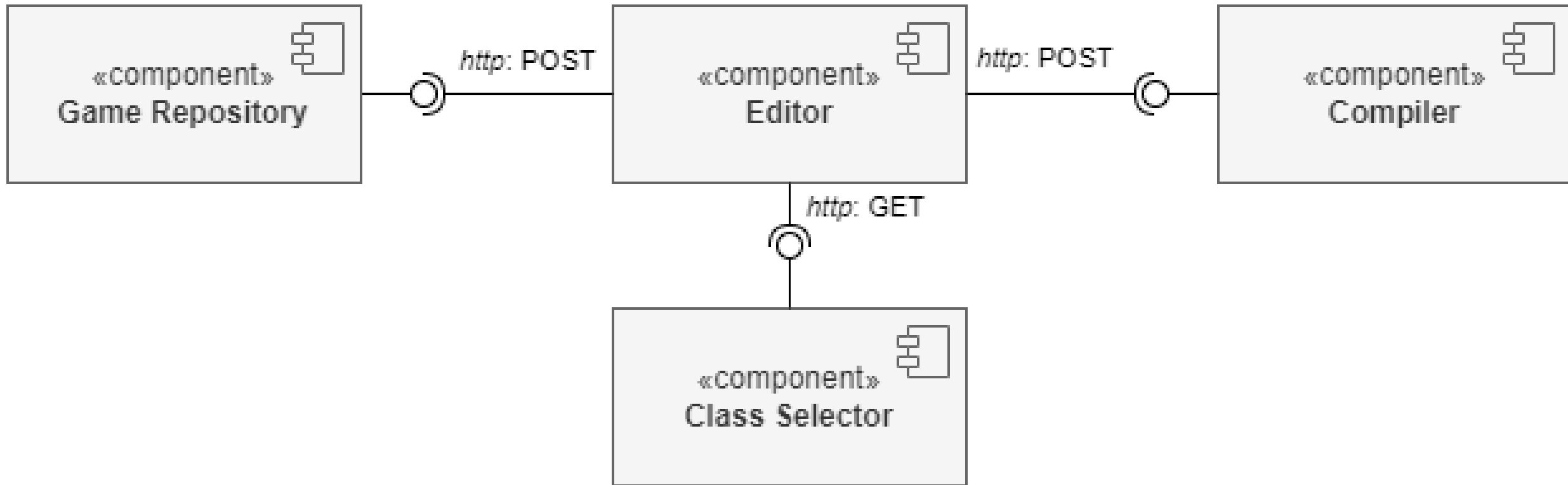
Artefatti

- Diagramma dei componenti di alto livello
- Diagramma dei componenti di basso livello
- Sprint Backlog
- Implementazione della funzionalità del bottone «Compile and Execute»
- Implementazione della funzionalità del bottone «Save Test»
- Implementazione della funzionalità del bottone «Download Test»
- Implementazione della funzionalità di Auto-completamento
- Implementazione della funzionalità di Search & Find
- Implementazione della funzionalità di Auto-Closing-Brackets
- Implementazione della funzionalità di Auto-indentazione
- Implementazione della funzionalità di Same Color Parenthesis

Artefatti Futuri

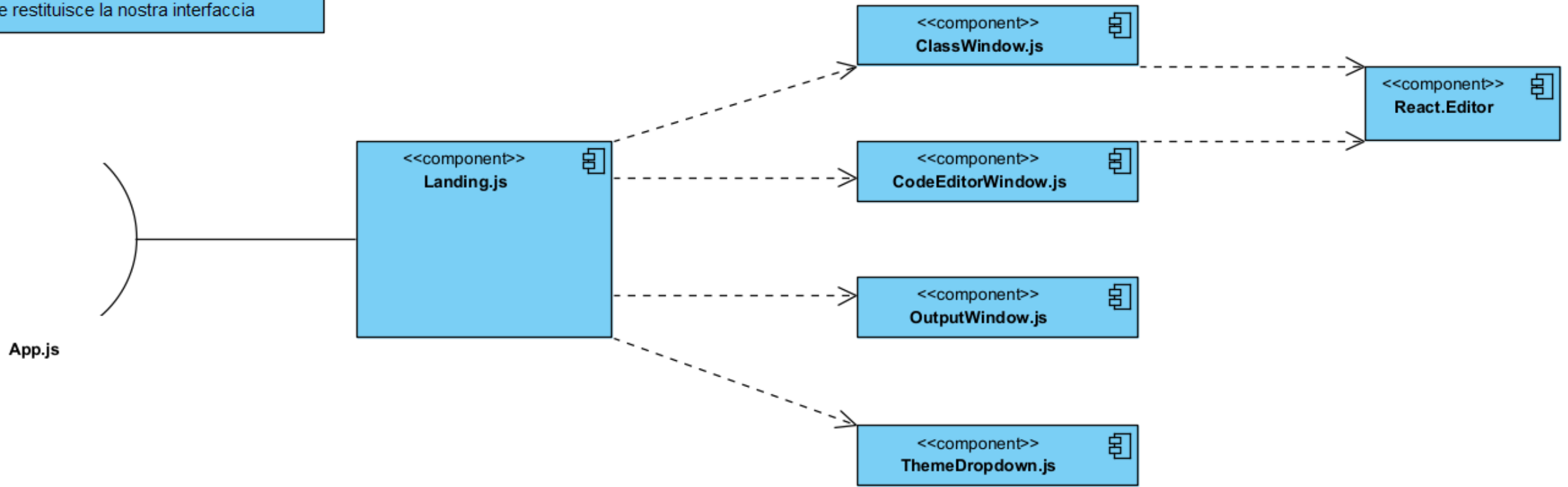
- Rifinitura della documentazione
- Rifinitura del codice

High Level Component Diagram



Low Level Component Diagram

Tali Componenti sono moduli usati per implementare il componente principale che restituisce la nostra interfaccia



Save Test

Download Test

Oceanic Next



Compile and Execute

```
1
2 import static org.junit.Assert.assertEquals;
3 import org.junit.Test;
4
5 public class AppTest{
6     @After
7     public void tearDown(){
8
9     }
10
11     @AfterClass
12     public static void tearDownClass(){
13
14     }
15
16     @Before
17     public void setUp(){
18
19     }
20
21     @BeforeClass
22     public static void setUpClass(){
23
24     }
25
26     @Test
27     public void test(){
28         assertEquals(2,1+1);
29     }
30 }
31
```

Class Under Test

```
1 package com.mkyong.examples;
2 public class ClassUnderTest {
3     public int add(int a, int b) {
4         return a + b;
5     }
6 }
```

Output

Sprint Backlog

ID	AS A ...	I WANT TO ...	SO THAT ...	PRIORITY	SPRINT	STATUS
001	Utente	Inserire codice in una finestra di editing	Poter scrivere la classe di test	High	2	Complete
002	Utente	Apportare modifiche	Poter modificare il codice nella finestra di editing	High	1	Complete
003	Sviluppatore	Aggiungere funzionalità di presentazione	Facilitare l'utente nell'attività di editing	Medium	3	Complete
004	Utente	Salvare file di testo creati	Poter mantenere la persistenza dei test scritti	Medium	2	Complete
005	Utente	Usufruire di un template precaricato	Essere facilitato nell'attività di coding	Low	1	Complete
006	Sviluppatore	Presentare una finestra editabile	Permettere all'utente di scrivere codice	High	1	Complete
007	Utente	Selezionare un tema di visualizzazione	Scegliere la grafica dell'editor	Low	1	Complete
008	Utente	Visualizzare la classe da testare	Per avere un riferimento nella scrittura del test e visualizzare la coverage	Medium	2	Complete
009	Utente	Compilare e lanciare il test	Avere il risultato di esecuzione	Medium	1	Complete
					0	
					0	
					0	
					0	
					0	
TOTAL					14	

Risultati ottenuti e sviluppi futuri

- Cosa abbiamo ottenuto a valle dell'iterazione ?
 - Modello finale dell'interfaccia di editing
 - Competenza delle tecnologie utilizzate
 - Simulazione dell'interazione con gli altri componenti
- Come intendiamo procedere nella prossima iterazione?
 - Rifinitura della documentazione:
 - Revisione dei diagrammi di analisi
 - Sviluppo dei diagrammi di progettazione
 - Rifinitura del codice:
 - Aggiunta di funzionalità all'editor (syntax highlighting)
 - Pulizia del codice