

Corso di Software Architecture Design

# Presentazione dei risultati della Iteration Review n. 2

9 Maggio 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: Creazione di un primo prototipo (statico)
- O2: Padronanza dei framework più complessi
- O3: Produzione e raffinamento della documentazione (diagrammi)
- O4: Realizzazione della prima storia utente con priorità alta

# Attività Svolte

## *Attività di ingegneria del software avviate o svolte*

- Raffinamento dei requisiti
- Sviluppo di Prototipi
- Progettazione preliminare
- Sviluppo di codice
- Raffinamento dell'interfaccia
- Sviluppo dei diagrammi di sequenza
- Definizione delle dipendenze con gli altri componenti

## *Altre attività organizzative/ di supporto al team svolte*

- Riunioni di gruppo tramite MS Teams
- Riunioni in presenza
- Organizzazione delle successive iterazioni
- Approfondimento della documentazione relativa alle tecnologie utilizzate

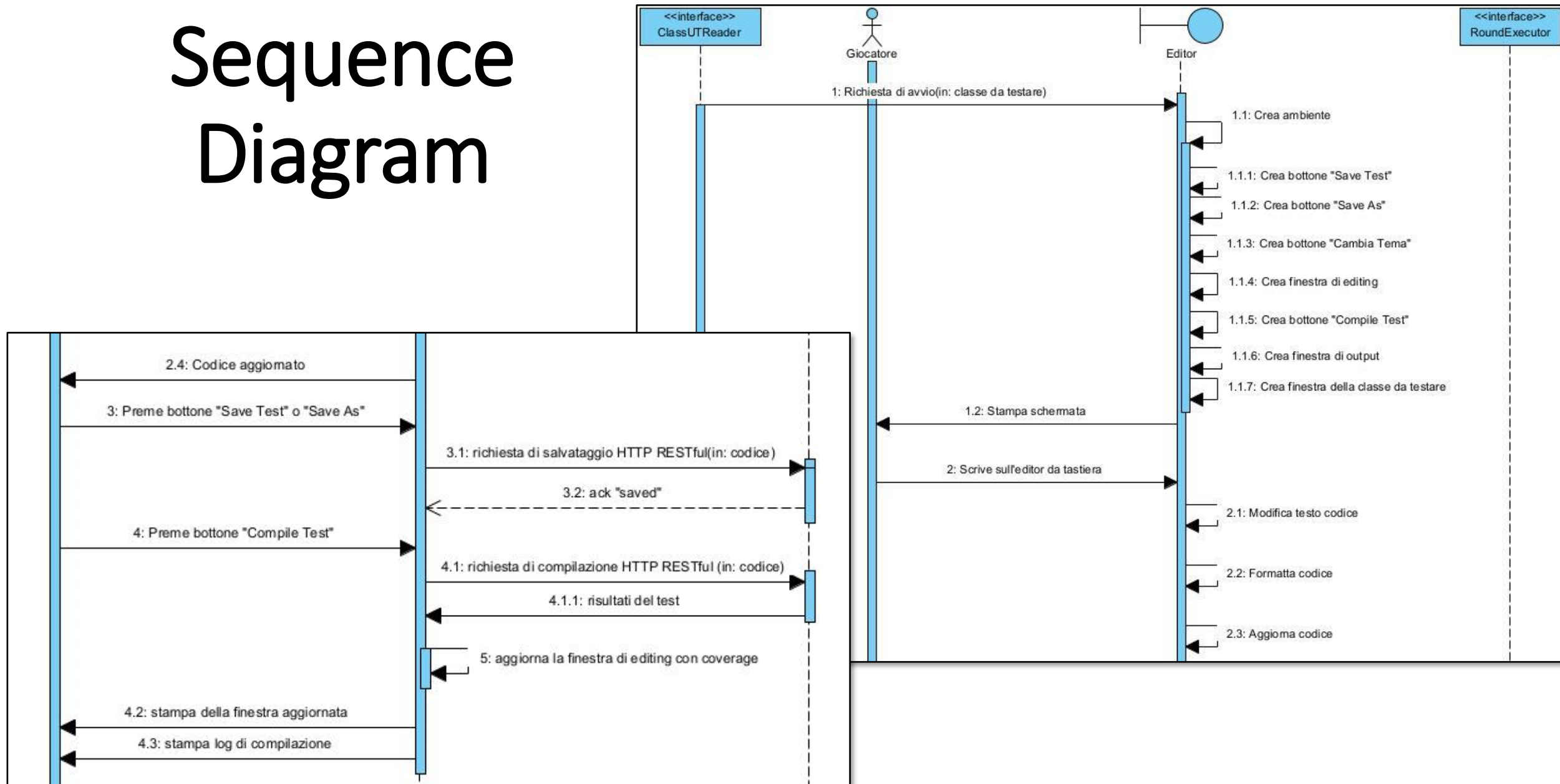
# Artefatti

- Diagramma di sequenza
- Prototipo dell'interfaccia
- Diagramma dei componenti
- Sprint Backlog

# Artefatti Futuri

- Rifinitura del codice
- Implementazione della logica relativa ai bottoni
- Implementazione delle successive storie utente

# Sequence Diagram



Save Test

Save as

Oceanic Next



Compile and Execute

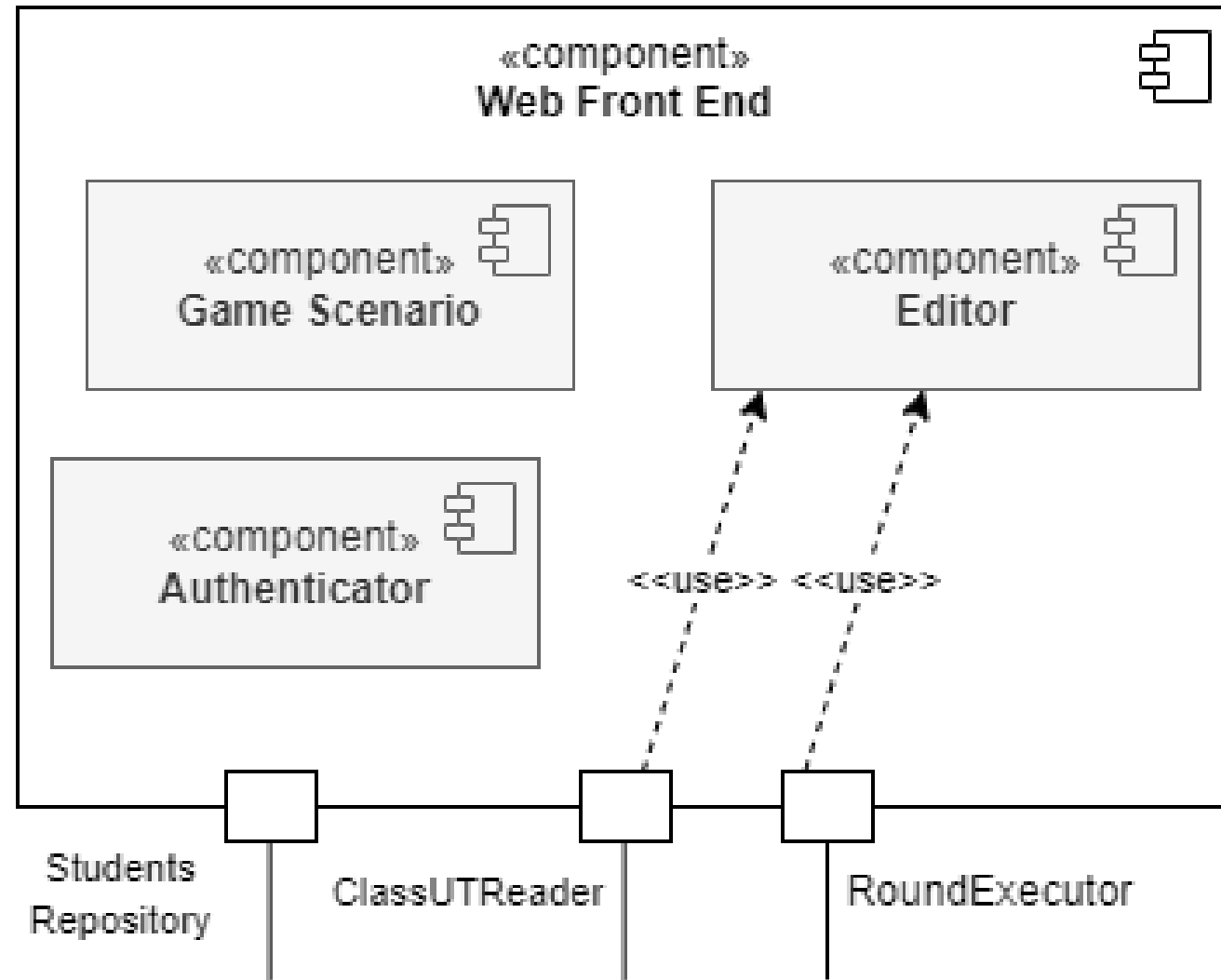
```
1
2 import static org.junit.Assert.assertEquals;
3 import org.junit.Test;
4
5 public class AppTest{
6     @After
7     public void tearDown(){
8
9     }
10
11     @AfterClass
12     public static void tearDownClass(){
13
14     }
15
16     @Before
17     public void setUp(){
18
19     }
20
21     @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
2 import java.util.ArrayList;
3 import java.util.List;
4
5 public class ByteArrayHashMap<T>
6 {
7     /**
8      * The default initial capacity - MUST be a power of
9      */
10     static final int DEFAULT_INITIAL_CAPACITY = 16;
11
12     /**
13      * The maximum capacity, used if a higher value is im
14      * by either of the constructors with arguments.
15      * MUST be a power of two <= 1<<30.
16      */
17     static final int MAXIMUM_CAPACITY = 1 << 30;
18
19
20     static final float DEFAULT_LOAD_FACTOR = 0.75f;
```

## Output

# Component Diagram



# Sprint Backlog

ID	USER STORY	TASKS	STATUS	EFFORT
002	Apportare modifiche al codice nella finestra di editing	Creare una finestra di editing	Complete	2
		Rendere la finestra editabile	Complete	
005	Usufruire di un template precaricato	Scrivere il template	Complete	1
		Caricarlo sulla finestra di editing	Complete	
006	Presentare una pagina di interfaccia	Creare una finestra di editing	Complete	5
		Creare i bottoni	Complete	
		Creare la finestra contenente la classe da testare	Complete	
		Creare la finestra di output	Complete	
008	Selezionare un tema di visualizzazione	Selezionare una lista di temi	Complete	1
		Implementare un menù di selezione	Complete	





# Risultati ottenuti e sviluppi futuri

- Cosa abbiamo ottenuto a valle dell'iterazione ?

- Primo prototipo dell'interfaccia di editing
- Maggiore padronanza delle tecnologie da utilizzare
- Visione globale dell'interazione con altri componenti

- Come intendiamo procedere nella prossima iterazione?

- Raffinamento del prototipo:
  - Implementazione delle funzionalità del bottone «Compile» (coverage)
- Raffinamento della visualizzazione della classe da testare
- Studio e approfondimento delle tecnologie per la generazione della coverage