



# Università degli Studi di Salerno

Corso di Ingegneria del Software  
Classe 1 Resto 0  
Corso di Laurea in Informatica  
A.A. 2022/23

## Quiad Rationale Management Document

Versione 1.0  
11/11/2022



**QUIAD**  
Family Tree

**Partecipanti al progetto e scriventi**

Nome	Matricola
Di Pasquale Valerio	0512110638
Troisi Vito	0512109807

---

**Revision History**

Data	Versione	Descrizione	Autore
11/11/2022	1.0	Prima stesura RMD, indice, razionale dietro il trade-off "Doppio caching"	D.P.V. T.V.

## **Indice**

- 1. Introduzione
  - 1.1 Overview
  - 1.2 Riferimenti
- 2. Architettura del sistema proposto
  - 2.1 Trade-off "Doppio caching"
- 3. Glossario

## **Overview**

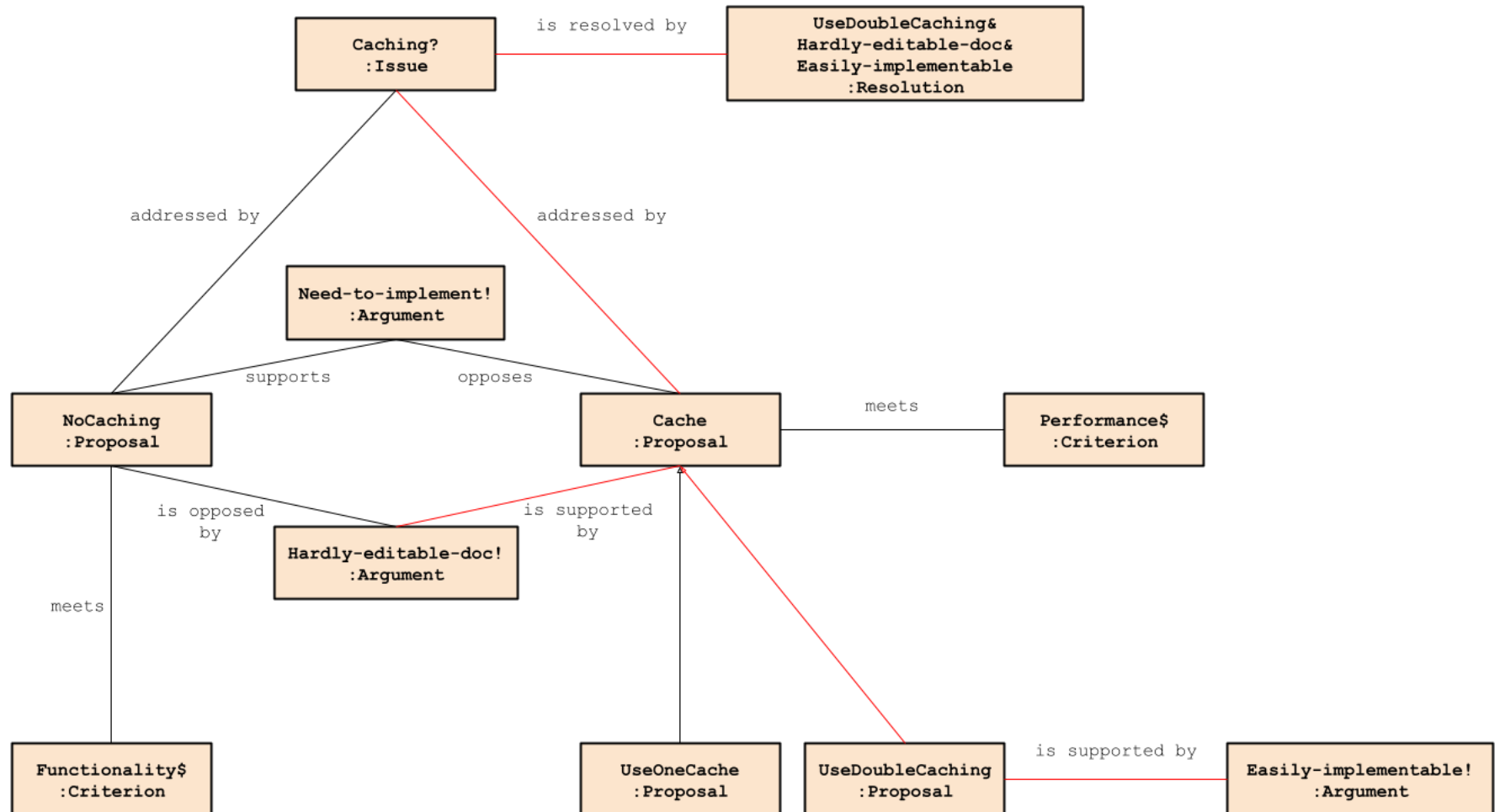
Il presente documento è atto a presentare il razionale dietro le principali scelte affrontate in fase di System Design. Saranno dunque evidenziate le issues affrontate, le alternative proposte in merito ad esse, i criteri che hanno guidato le scelte e le argomentazioni che hanno supportato la decisione finale.

Il razionale sarà espresso mediante gli opportuni diagrammi, i quali esibiranno anche la resolution proposta. Le argomentazioni saranno riportate in seguito a ciascun diagramma.

Prima di procedere nella lettura del documento in questione si raccomanda di leggere con attenzione la sezione Riferimenti ed il Glossario dei termini rilevanti.

## **Riferimenti**

Il presente documento riguarda il razionale dietro le scelte fatte in fase di progettazione del sistema Quiad ed esplicitate nel System Design Document. Si raccomanda dunque di osservare in quest'ultimo l'applicazione concreta delle risoluzioni alle issues proposte.

**Trade-off "Doppio caching"**

**Argomentazione "Need-to-implement"**

Come sarà probabilmente apparente, l'utilizzo di una tecnica di caching, o doppio caching, necessita dell'implementazione di tale tecnica, sul client e sul server.

**Argomentazione "Hardly editable-doc"**

Si osservi che un documento, se richiesto dall'utente, dev'essere stato approvato da un utente supervisore in quanto è stato reso disponibile alla visualizzazione. Osservando che i documenti sono di carattere storico, una ragionevole ipotesi potrebbe osservare che poco probabilmente un documento sarà soggetto a modifiche.

**Argomentazione "Easily-implementable"**

L'implementazione del meccanismo, sia esso lato client o server, consiste in una struttura dati che attui un'associazione chiave-valore di elementi (i.e. i documenti) a cui è associata una data scadenza (e.g. un'ora). Molte delle tecnologie utilizzate per l'implementazione di sistemi client/server sono provvisti di API che consentono l'uso del caching. La difficoltà di implementazione, è pertanto meno significativa del guadagno in termini di performance.

**Glossario**

<b>Quiad</b>	<p>Il nome del progetto e della piattaforma che ne concretizza i servizi. Consente la gestione del proprio albero genealogico e la ricerca di associata documentazione storica.</p> <p>E' latino, e significa "A chi"!</p>
<b>Issue/ Problematica</b>	<p>I problemi decisionali affrontati in fase di System Design. Richiedono l'analisi della alternative proposte.</p>
<b>Alternativa/ Proposal</b>	<p>Le differenti proposte di soluzione per un'assegnata issue. Possono, a loro volta, sollevare problematiche.</p>
<b>Criterion/ Criterio</b>	<p>Una metrica di giudizio della "bontà" di un'alternativa. Per comprendere al meglio i criteri, si raccomanda la lettura degli obiettivi di progettazione esposti nel System Design Document.</p>
<b>Argument/ Argomentazione</b>	<p>Un'arringa che supporta o contrasta un'alternativa. Le argomentazioni costituiscono l'approccio principale al razionale ivi presentato.</p>
<b>Resolution/ Risoluzione</b>	<p>La decisione di intraprendere una data alternativa proposta, così da chiudere (i.e. risolvere) una issue.</p>