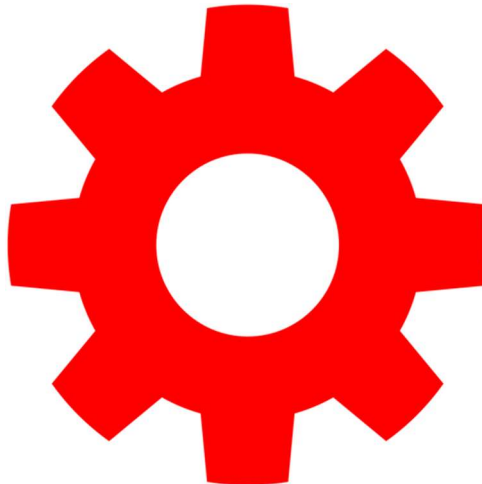


**ePADs**  
**System Design Document - SDD**  
**Versione 2.0**



# INDICE

<b>1.INTRODUZIONE .....</b>	<b>2</b>
<b>1.1 Purpose Of The System .....</b>	<b>2</b>
<b>1.2 Design Goals .....</b>	<b>3</b>
<b>1.3 Riferimenti .....</b>	<b>3</b>
<b>2.PROPOSED SOFTWARE ARCHITECTURE .....</b>	<b>4</b>
<b>2.1Subsystem Decomposition .....</b>	<b>4</b>

## 1. INTRODUZIONE

### 1.1 Purpose Of The System

Si vuole realizzare un tool, denominato ePADs, che sia capace di identificare Design

Pattern, che consiste nel riconoscimento di istanze di Design Pattern all'interno di sistemi software o applicazioni esistenti.

## 1.2 Design Goals

- **Performance:** Il tool dovrà rispondere velocemente; La latenza massima di attesa, per una risposta, non dovrà superare i 60 secondi.
- **Supportability:** Il tool dovrà essere suddiviso in vari package per permettere una facile modifica e aggiornabilità in futuro.
- **Implementation:** il developer inserirà da linea di comando la path, la jarPath dove si trova il contenuto da controllare e una serie di comandi che serviranno per effettuare tutte le funzioni del tool.
- **Gestione:** il software sarà gestito dal developer stesso.

## 1.3 Riferimenti

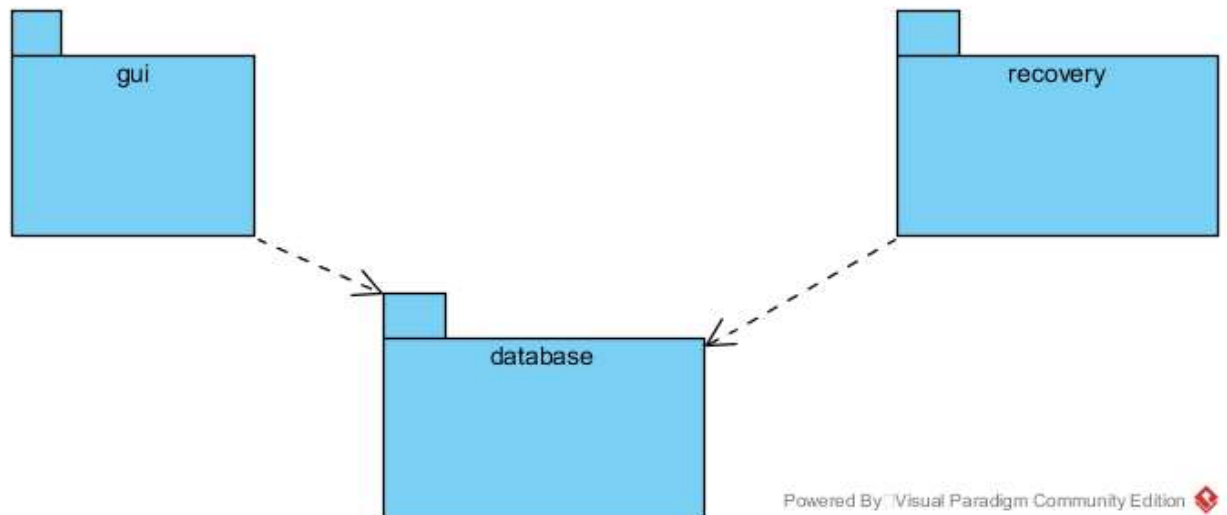
In questo documento si fa riferimento a:

- RAD\_Versione\_3.0

## 2. PROPOSED SOFTWARE ARCHITECTURE

La decomposizione in sottosistemi è stata effettuata seguendo la tecnica del partizionamento.

Il partizionamento, ha consentito, di classificare i vari sottosistemi in base allo stesso livello di astrazione.



I sottosistemi individuati all'interno del tool ePADs , inoltre, rispecchiano, in linea di massima, le funzionalità offerte dal sistema.

1. **gui**: fornisce il servizio per recuperare tutte le checkoptions inserite dal developer nel momento in cui viene lanciato il tool ePADs.
2. **Recovery**: fornisce i servizi per svolgere le varie fasi di estrazione dei design pattern.
3. **Database**: fornisce i servizi per recuperare i dati delle classi di design pattern.