**Cliente – Progetto**

**System Architecture Document**

|  |  |
| --- | --- |
| Preparato da | Inserire l’autore; indicare eventualmente anche il responsabile tecnico di Objectway che ha approvato |
| Versione | n.m; se in bozza, specificarlo. |
| Data | Inserire la data di preparazione/ultimo aggiornamento |
| Codice | SAD-[cod comm]-[abbrev comm]-n.m |

# **Indice**

[1. Premessa 3](#_Toc394932578)

[1.1. Responsabilità 3](#_Toc394932579)

[1.2. Evoluzione del Documento 3](#_Toc394932580)

[1.3. Elenco Documenti Allegati 3](#_Toc394932581)

[1.4. Abbreviazioni 3](#_Toc394932582)

[2. Aspetti generali 4](#_Toc394932583)

[2.1. Presentazione dell’architettura 4](#_Toc394932584)

[2.2. Suddivisione in sottoprogetti 4](#_Toc394932585)

[2.3. Requisiti e Vincoli 4](#_Toc394932586)

[3. La vista dei casi d’uso 5](#_Toc394932587)

[3.1. <Use Case Realization 1> 5](#_Toc394932588)

[4. La vista logica 6](#_Toc394932589)

[4.1. Introduzione 6](#_Toc394932590)

[4.2. <sottoprogetto 1> 6](#_Toc394932591)

[La struttura generale 6](#_Toc394932592)

[Aspetti generici dell’architettura 7](#_Toc394932593)

[<sottosistema/package 1> 7](#_Toc394932594)

[5. La vista dei processi 8](#_Toc394932595)

[5.1. <gruppo di processi 1> 8](#_Toc394932596)

[<processo 1> 8](#_Toc394932597)

[Processi esterni e comunicazioni tra processi 8](#_Toc394932598)

[6. La vista del dislocamento 9](#_Toc394932599)

[<Nodo 1> 9](#_Toc394932600)

[Collegamenti e Nodi esterni 9](#_Toc394932601)

[7. La vista dell’implementazione 10](#_Toc394932602)

[7.1. I sottosistemi del [sotto]progetto <nome> 10](#_Toc394932603)

[7.1.1. <Livello 1> 10](#_Toc394932604)

[7.1.1.1. Componenti commerciali 10](#_Toc394932605)

[8. Vista dei dati 11](#_Toc394932606)

[9. Descrizione dei sistemi e della rete 12](#_Toc394932607)

Indice delle Figure

[Figure 1 – nome figura 4](#_Toc394933731)

[Figure 1: sequence o collaboration diagram 5](#_Toc394933732)

[Figure 3: Sottosistemi Applicativi 6](#_Toc394933733)

[Figure 4: Esempio di attività dei sottosistemi 6](#_Toc394933734)

[Figure 5: Processi del gruppo applicativo 8](#_Toc394933735)

[Figure 6: Comunicazione tra i processi del gruppo e altri gruppi 8](#_Toc394933736)

[Figure 7: Schema dei nodi e delle connessioni 9](#_Toc394933737)

[Figure 8: Interdipendenze tra i sottosistemi applicativi 10](#_Toc394933738)

Indice delle tabelle

[Table 1 – nome tabella 5](#_Toc394933739)

# Premessa

## Responsabilità

Il presente documento e ogni sua revisione devono essere approvati dal Responsabile Tecnico OW e dal Responsabile Tecnico delCliente.

## Evoluzione del Documento

*[Questa sezione stabilisce se esistono dei momenti del progetto in cui viene pianificata una revisione del SAD.]*

## Elenco Documenti Allegati

|  |  |
| --- | --- |
| **Codice Documento** | **Titolo Documento** |
|  |  |

## Abbreviazioni

*le informazioni in questa sezione possono fare riferimento a un eventuale Glossario di progetto.*

|  |  |
| --- | --- |
| **Abbreviazione** | **Significato** |
|  |  |

# Aspetti generali

## Presentazione dell’architettura

*[descrivere quale architettura software è stata scelta per il progetto in corso, e come viene rappresentata. Vengono enumerate quali tra le viste del RUP (Use-Case, Logical, Process, Deployment, Implementation) sono necessarie e per ciascuna si spiega che tipo di elementi contengono i suoi modelli.]*

*[Poichè la vista dei casi d’uso (se prevista) è presentata in dettaglio nel documento SRD, nel presente documento si prende in considerazione la loro realizzazione negli scenari o diagrammi di collaborazione (nel RUP vedi Analysis and Design: Artifacts: use-case realization), in termini di oggetti e classi che poi saranno organizzati in diagrammi nella vista logica (vedi Rational Rose). Se per motivi contingenti non è possibile o si decide di che non è necessaria questa parte, si fa riferimento al SRD e si passa direttamente alla vista logica.]*

*Esempio immagine*



Figure 1 – nome figura

## Suddivisione in sottoprogetti

Se il progetto è complesso può essere utile considerare una suddivisione in sottoprogetti, caratterizzati sia dalle tecnologie che dalle metodologie specifiche che dovranno essere utilizzate per realizzarli.

Nel resto del documento, comunque, si da risalto alla prospettiva delle viste; in ciascuna vista vengono presentati gli aspetti relativi ai vari sottoprogetti e le relazioni tra loro. A seconda delle esigenze si può fare un unico SAD oppure un SAD per ciascun sottosistema.

## Requisiti e Vincoli

*[descrivere, oppure fare riferimento se contenuti in altri documenti, i requisiti software e gli obiettivi che hanno un impatto significativo sulla architettura, per esempio, sicurezza, riservatezza, uso di un prodotto off-the-shelf, portabilità, distribuzione, riuso. Indicare anche i vincoli speciali che potrebbero applicarsi: strategie di design e implementazione, tools di sviluppo, struttura del team, tempistica, codice preesistente, etc.].*

*[Specificare eventuali requisiti legali o regolamentari da rispettare; per esempio normativa privacy, normativa sulla firma digitale e il documento elettronico].*

# La vista dei casi d’uso

*[elencare, o fare riferimento se contenuti in altri documenti, gli scenari identificati a partire dal modello dei casi d’uso, se rappresentano qualche funzionalità significativa o centrale del sistema finale, o se hanno un vasto impatto architetturale, o se illustrano una specifica delicata parte dell’architettura.*

*La descrizione dettagliata dei casi d’uso del sistema è presentata nel SRD. Per questo motivo non si ripete la descrizione di Attori e Casi d’Uso.]*

*Esempio di Tabella*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 1 – nome tabella

## <Use Case Realization 1>

*[illustrare come il software funziona effettivamente, fornendo alcune realizzazioni di casi d’uso o scenari selezionati, e spiegare come i vari elementi del modello di design contribuiscono alle loro funzionalità.]*

Figure 1: sequence o collaboration diagram

# La vista logica

*[descrivere le parti architetturalmente significanti del design model, come la sua decomposizione in sottosistemi e packages. Per ogni package significativo, dare la sua decomposizione in classi e classi utility. Introdurre le classi significative e descrivere le loro responsabilità, così come le più importanti relazioni, operazioni e attributi.]*

## Introduzione

*[descrivere la decomposizione d’alto livello del modello di design in termini della sua gerarchia di package e livelli (layers)]*

Per esempio si potrebbe decidere di separare il sottoprogetto Applicativi dal sottoprogetto Database. Il primo comporta la realizzazione dei Casi d’Uso più orientati al dominio applicativo ed avrà come prodotto un insieme di programmi rivolti all’utente finale. Questi programmi sfrutteranno i dati messi a disposizione attraverso il sottoprogetto Database, che comporta la realizzazione della base di dati vera e propria, e degli strumenti necessari alla sua amministrazione. In questo senso il sottoprogetto Database è maggiormente orientato alla realizzazione dei Casi d’Uso rivolti all’amministrazione del sistema.

L’interfaccia tra i due sottoprogetti è costituita da...

## <sottoprogetto 1>

*[Se il documento non prevede altri sottoprogetti eliminare un livello di headings e scalare “in su” i successivi]*

### La struttura generale

Ad esempio:

La struttura generale del sistema è composta da tre livelli o sottosistemi (package), secondo uno schema concettuale consolidato. Ognuno dei livelli è responsabile per la fornitura di un tipo di servizi, e comunica soltanto per richiedere servizi dal livello immediatamente inferiore, o per rispondere a richieste del livello superiore. I tre livelli sono:

Figure 3: Sottosistemi Applicativi

*Persistenza*, ovvero i meccanismi volti al recupero di informazioni dal Database ed al loro eventuale aggiornamento;

*Modello*, cioè la rappresentazione delle entità proprie del dominio applicativo, come ad esempio Clienti e Contratti, delle relazioni che le legano, ad esempio ogni Contratto può avere dei Movimenti, e delle operazioni e dei calcoli che possono essere effettuati su di esse, come ad esempio il calcolo del rendimento;

*Presentazione*, vale a dire le funzionalità volte a presentare le informazioni all’utente, in forma visuale o stampata.

A questi sottosistemi se ne aggiunge un quarto, in un certo senso trasversale, detto Utilità, che fornisce servizi generici agli altri tre.

Figure 4: Esempio di attività dei sottosistemi

Il ruolo concreto dei diversi sottosistemi può essere chiarito esaminando, a titolo di esempio, uno scenario derivato dal Caso d’Uso della Ricerca del Cliente, come descritto nel Diagramma 3. …

### Aspetti generici dell’architettura

[Opzionale]

Per garantire un giusto grado di uniformità tra i diversi elementi del sottoprogetto è utile identificare un insieme di elementi architetturali e di linee guida comuni a tutti i sottosistemi. Ad esempio:

* La definizione dei tipi di dati di base;
* La specifica di elementi strutturali, o pattern, ricorrenti.

### <sottosistema/package 1>

*[per ogni package significativo, includere una sotto-sezione con il suo nome, una breve descrizione, e un diagramma con le classi e i package significanti contenuti nel package.*

*Per ogni classe significativa nel package, includere nome, breve descrizione, e, opzionalmente una descrizione delle principali responsabilità, operazioni e attributi.]*

# La vista dei processi

*[descrivere la decomposizione del sistema in processi (singoli threads di controllo) e gruppi di processi. Organizzare la sezione per gruppi di processi che comunicano tra loro o interagiscono. Descrivere i principali modi di comunicazione tra processi, come passaggio di messaggi, interrupts, rendezvous.]*

La Vista dei Processi costituisce la controparte dinamica della Vista Logica, ovvero mentre la seconda presenta elementi statici, che spesso corrispondono a entità passive del mondo reale, la prima descrive le modalità con il quale il sistema agisce, normalmente a seguito dell’interazione con l’utente.

Anche in questo caso la distinzione tra i sottoprogetti del sistema è rilevante, poiché realizzano package di casi d’uso diversi e interagiscono con attori differenti. A ciascun sottoprogetto corrisponde un gruppo di processi.

## <gruppo di processi 1>

*[dare un titolo significativo a ogni sottosezione]*

Figure 5: Processi del gruppo applicativo

### <processo 1>

*[descrizione del processo]*

### Processi esterni e comunicazioni tra processi

*[Descrivere i principali modi di comunicazione tra processi, come passaggio di messaggi, interrupts, rendezvous.]*

Figure 6: Comunicazione tra i processi del gruppo e altri gruppi

# La vista del dislocamento

*[descrivere uno o più configurazioni di reti fisiche (hardware) su cui il software è installato ed eseguito. È una vista del Deployment Model. Come minimo per ogni configurazione si dovrebbero indicare i nodi fisici (computers, CPUs) che eseguono il software e le loro interconnessioni (bus, LAN, point-to-point,etc.). Includere anche il mappaggio dei processi della Process View sui nodi fisici.]*

Mentre le viste precedenti descrivono la struttura del sistema dal punto di vista concettuale, la Vista del Dislocamento descrive alcuni aspetti concreti della soluzione, ed in particolare la distribuzione dei processi sui diversi dispositivi di calcolo. Il sistema *<nome>* è dislocato su due tipi di sistemi, o nodi, ovvero Client e Server Database, su cui eseguono i processi del progetto, i quali interagiscono con alcuni sistemi esterni, su cui eseguono i processi esterni descritti nel Capitolo 5.

Figure 7: Schema dei nodi e delle connessioni

### <Nodo 1>

*[descrizione del nodo e processi ad esso allocati]*

### Collegamenti e Nodi esterni

*[collegamenti tra nodi interni e esterni al sistema]*

# La vista dell’implementazione

*[descrivere la struttura complessiva del modello dell’implementazione, la decomposizione in layer e sottosistemi del software, e qualsiasi componente architetturale significativo.]*

La vista dell’Implementazione descrive gli artefatti che comporranno materialmente il progetto rilasciato, suddivisi in base al sottoprogetto di riferimento. Tali artefatti costituiranno la realizzazione concreta degli elementi descritti nella Vista Logica, la cui esecuzione darà luogo ai processi descritti nella vista corrispondente.

## I sottosistemi del [sotto]progetto <nome>

*[nominare e definire i vari layer e i loro contenuti, le regole che governano l’inclusione in un certo layer, e i confini tra layers. Un diagramma a componenti mostra le relazioni tra layers. ]*

Figure 8: Interdipendenze tra i sottosistemi applicativi

* + 1. <Livello 1>

*[per ogni layer, includere una sottosezione con il suo nome, un elenco dei sottosistemi collocati in tale layer e un diagramma a componenti.]*

* + - 1. Componenti commerciali

*[elenco di eventuali componenti acquistati]*

# Vista dei dati

*[Opzionale]*

*[una descrizione della prospettiva dei dati persistenti del sistema. Opzionale se ci sono pochi dati persistenti o se non ce ne sono, oppure se il passaggio dal Design Model al Data Model è banale.]*

# Descrizione dei sistemi e della rete

*[Obbligatoria per i progetti IPS e quelli in cui l’infrastruttura è gestita da Objectway]*

*[Riportare la descrizione dei sistemi coinvolti:*

* *server e caratteristiche (se virtuali, se in load balancing, se in cluster)*
* *apparati di rete e caratteristiche (se ridondati, con quali prestazioni, banda di rete)*
* *sistemi operativi, middleware e DBMS, inclusa la loro versione*
* *servizi di appoggio*
* *includere eventuali siti di Disaster Recovery*
* *includere eventuali connessioni con altre parti e i servizi coinvolti*
* *includere sistemi di backup*
* *includere ambienti di test e collaudo]*