

Titolo del Progetto

Membri del team di sviluppo:
nome cognome matricola
nome cognome matricola
nome cognome matricola

Sommario:

Sommario

Abstract.....	3
Analisi dei requisiti.....	4
Requisiti del sistema.....	4
Analisi del dominio	4
Analisi dei requisiti.....	4
Analisi dei rischio	4
Descrizione delle interfacce grafiche (OPZIONALE).....	5
Analisi del problema	6
Modello del dominio	7
Architettura logica	7
Struttura.....	8
Interazione	8
Comportamento	8
Piano di Lavoro	8
Piano del Collaudo	9
Progetto	10
PROGETTAZIONE ARCHITETTURALE	10
Requisiti non funzionali	10
Scelta dell'architettura	10
Scelte tecnologiche (opzionale)	10
PROGETTAZIONE DI DETTAGLIO.....	11
Struttura.....	11
Interazione	11
Comportamento	11
PROGETTAZIONE DELLA PERSISTENZA.....	11
PROGETTAZIONE DEL COLLAUDO	11
PROGETTAZIONE DEL DEPLOYMENT.....	11
Progettazione del deployment per la sicurezza.....	11
Deployment del sistema.....	11
Implementazione	12
Scelte tecnologiche (opzionale)	12
Scelte Implementative	12
Collaudo	12
Deployment (opzionale).....	12

Abstract

Breve descrizione del progetto

Analisi dei requisiti

Requisiti del sistema

In questa sezione vanno indicati in modo chiaro tutti i requisiti del sistema. Nel nostro processo di sviluppo rappresenta la fase finale della Raccolta dei Requisiti. Al fine di supportare la tracciabilità ci si può avvalere di una tabella come quella sottostante

Id. Requisito	Requisito	Tipo
Numero identificativo univoco	Requisito	Funzionale/dominio/non funzionale

Analisi del dominio

- Qua va messa la parte relativa all'analisi del dominio applicativo e viene costruita la prima versione del **vocabolario** partendo dai "sostantivi" che si trovano nei requisiti. Se lo si ritiene opportuno è possibile mettere in questa parte i requisiti "colorando" i sostantivi presenti nella specifica dei requisiti come nell'esempio del Villaggio Turistico visto a lezione.
- Se si utilizzano software di terze parti vanno indicati in questa sezione (attenzione che poi l'analisi ai morsetti vera e propria dello strumento va fatta nella fase di analisi del problema).
- Se si usa come base di partenza una applicazione esistente che si intende migliorare o modificare, va indicata in questa sezione.
- In questa sezione è anche possibile fare un confronto con altri sistemi che operano nello stesso dominio per valutarne punti di forza e di debolezza.

Analisi dei requisiti

Qua viene fatta l'analisi dei requisiti e vengono identificati i requisiti funzionali. Si analizzano nel dettaglio i "verbi" presenti nella specifica dei requisiti. Porre particolare attenzione nella modellazione dei ruoli (attori) che interagiranno con il sistema.

Per ogni requisito inserito nella tabella dei requisiti va fatta una dettagliata analisi.

Va inserito il **modello dei casi d'uso e i relativi scenari**

Analisi del rischio

Seguiamo l'analisi del rischio vista nella parte della sicurezza, quindi in questa sezione andranno indicati:

Tabella Valutazione dei Beni

Bene	Valore	Esposizione
nome	Valore del bene	Tipo di esposizione (finanziaria, di immagine, etc.)

Tabella Minacce/Controlli

Minaccia	Probabilità	Controllo	Fattibilità
Tipo di	Probabilità di questa	Controllo 1	Quanto costa il controllo

minaccia	minaccia	Controllo 2	Quanto costa il controllo
----------	----------	-------------	---------------------------

Analisi Tecnologica della Sicurezza

Tecnologia	Vulnerabilità
Tipo di tecnologia da usare	Vulnerabilità della tecnologia

Vanno poi realizzati i **Security Use Case** e i **Misuse Case** con relativi scenari

Infine vanno chiaramente indicati i requisiti di sicurezza che si evincono dall'analisi del rischio.

Descrizione delle interfacce grafiche (OPZIONALE)

Se necessario è possibile delineare già con il cliente l'aspetto delle interfacce grafiche. Specialmente nel caso di applicazioni gestionali delineare le interfacce aiuta a capire come "raggruppare" i dati nelle maschere di immissione.

Analisi del problema

In questa sezione va specificata tutta l'analisi del problema: si parte dall'analisi dei requisiti (requisiti funzionali e non funzionali finiscono nello stesso calderone e li analizzo tutti) e si va ad investigare nel dettaglio il problema.

In questa parte vanno analizzati anche ai morsetti eventuali strumenti esterni con cui si deve interagire, eventualmente si studiano anche i sensori e si analizza nel dettaglio la struttura delle interfacce discusse con il cliente al fine di analizzare il flusso dei dati nel sistema.

Gli artefatti che vanno prodotti al termine di questa fase sono: **architettura logica** del problema, il **piano del collaudo** e il **piano di lavoro**.

Vanno riempite le seguenti tabelle

Analisi Documento dei Requisiti: Analisi delle Funzionalità

Tabella Funzionalità

Funzionalità	Tipo	Grado Complessità	Requisiti Collegati
nome	Tipo di funzionalità	Complessa/semplice	Requisiti a cui è collegata la funzionalità

Tabella Informazioni/Flusso (1 per ogni funzionalità)

Informazione	Tipo	Livello protezione / privacy	Input/output	Vincoli
nome	Semplice/complessa	Protezione bassa/media/alta/altissima		Vincoli sui valori che possono essere assunti

Analisi Documento dei Requisiti: Analisi dei Vincoli

Tabella Vincoli

Requisito	Categorie	Impatto	Funzionalità
nome	Categorie requisito		Elenco funzionalità coinvolte

Analisi Documento dei Requisiti: Analisi delle Interazioni

Tabella Maschere

Maschera	Informazioni	Funzionalità
Nome maschera	Quale informazioni mostra/richiede	Elenco funzionalità coinvolte

Tabella Sistemi Esterni

Sistema	Descrizione	Protocollo di Interazione	Livello di Sicurezza
Nome sistema esterno	descrizione	Spiegare le API messe a disposizione	Livello di sicurezza fornito

Analisi Ruoli e Responsabilità

Tabella Ruoli

Ruolo	Responsabilità	Maschere	Riservatezza	Numerosità
Nome ruolo	Elenco responsabilità	Elenco maschere	Grado di riservatezza richiesta	Numero di persone che giocano il ruolo

Tabella Ruolo-Informazioni (una per ogni ruolo)

Informazione	Tipo di Accesso
Nome informazione	Lettura/scrittura

Scomposizione del Problema

Tabella Scomposizione Funzionalità

Funzionalità	Scomposizione
nome	Elenco funzioni scomposte

Tabella Sotto-Funzionalità (una per ogni funzione)

Sotto-funzionalità	Sotto-funzionalità	Legame	Informazioni
nome	nome	Tipo di legame	Elenco informazioni che di scambiano

Modello del dominio

Riportare e commentare il diagramma delle classi del dominio (i dati) del problema.

Si parte dal vocabolario e si crea da quello il **modello dei dominio** che di solito rappresenta il nucleo iniziale della struttura dati del nuovo sistema (tipicamente sarà uno dei “package” che vanno nell’architettura logica).

Architettura logica

L’architettura logica del problema viene analizzata da tre diverse prospettive: struttura, interazione e comportamento.

ATTENZIONE: le tre viste devono essere coerenti tra loro, esempio: se due classi sono poste in un qualche tipo di relazione allora ci deve essere un diagramma di sequenza che ne specifica l’interazione.

Struttura

Generalmente la parte strutturale dell'architettura logica viene presentata a diversi livelli di dettaglio usando diversi diagrammi UML

- **Diagramma dei package** → è il primo che viene inserito e mostra come è logicamente suddiviso il nostro problema. ATTENZIONE: qua siamo ancora nel dominio del problema, la suddivisione in package deve seguire la logica del problema e NON quella della soluzione. Le dipendenze tra i package sono dipendenze logiche che potrebbero essere sovvertite in fase di progettazione!!!
- **Diagramma delle classi** → per ogni package vanno poi inserite le classi (che in questa fase rappresentano ancora entità astratte del problema, NON sono le classi dell'implementazione per ora!!) con gli attributi che si possono identificare dai requisiti e specificati nelle tabelle sopra. Attenzione a seguire anche le specifiche di sicurezza...ogni ruolo vede solo ed esclusivamente le informazioni che gli competono. Le operazioni che compaiono sono solo quelle che possiamo dedurre dai requisiti e che quindi sono riconducibili ai servizi richiesti al sistema.

Interazione

In questa parte va prima analizzata in dettaglio l'interazione:

- tra le varie parti del sistema definite nella struttura
- tra il sistema e l'utente esterno (basatevi sui casi d'uso per definirla bene)
- tra il sistema e gli eventuali strumenti software esterni
- tra il sistema e gli eventuali componenti hardware (se avete a che fare con dei sensori va anche analizzato cosa si deve fare in caso di smarrimento di pacchetti di dati etc..)

Vanno prodotti svariati **diagrammi di sequenza** per mostrare come avviene l'interazione specificando la sequenza dei "messaggi" scambiati. Fate attenzione che i messaggi sincroni hanno bisogno sempre del ritorno e che i dati vanno recuperati dalle classi del dominio, i controller non se li possono inventare. Attenzione anche a mantenere la giusta sequenza temporale!!

Comportamento

Tipicamente in questa parte vengono riportati i diagrammi di stato e/o di attività delle entità definite nella parte strutturale (attenzione che non tutte saranno dotate di uno stato interno) in modo da chiarire bene come si comportano in base agli eventi che agli input che ricevono dall'esterno. Attenzione che spesso viene confuso il diagramma di stato con quello delle attività!

Piano di Lavoro

Il piano di lavoro esprime l'articolazione delle attività in termini di risorse umane (a chi assegno ciascuna parte del progetto e del successivo sviluppo), temporali (che tempi di consegna ci sono per ogni parte), di elaborazione etc. necessarie allo sviluppo del prodotto in base ai risultati dell'analisi del problema. Inoltre nel piano di lavoro va indicato chiaramente cosa ci sarà nella versione del prototipo che verrà mostrato all'esame. Se non ci sono indicazioni ci si aspetta che nel prototipo ci saranno tutte le funzionalità.

Piano del Collaudo

Il piano di collaudo definisce l'insieme dei risultati attesi (congruenti con i requisiti) da ogni parte (sottosistema, componente, etc.) definito nell'architettura logica in base al comportamento osservabile di ciascuna parte in relazione a specifiche sollecitazioni stimolo-risposta prevista.

Un buon modo per impostare il piano di collaudo è quello di fornire i test delle classi che sono state definite nell'architettura logica, questo permette anche di pianificare già in questa fase i test di integrazione dei vari sottosistemi in modo da agevolare il lavoro nella fase di progetto cercando di evitare brutte sorprese quando si fa l'integrazione delle sotto-parti.

Per C# potete usare l'ambiente NUnit (<http://nunit.org/>) mentre per Java potete usare JUnit.

Progetto

Lo scopo della fase di progettazione consiste nel raffinamento dell'architettura logica del problema sino ad ottenere l'architettura del sistema che deve essere sviluppato.

Verranno quindi inserite tutte le nuove classi che permettono di risolvere i problemi identificati in fase di analisi, saranno progettati tutti i protocolli di interazione e progettato il comportamento delle nuove classi inserite. Saranno inoltre raffinate le classi delineate nell'architettura logica inserendo eventualmente nuovi attributi e nuove operazioni.

In questa fase potrebbe anche essere necessario modificare le dipendenze logiche espresse nell'analisi al fine di migliorare lo scambio dei dati.

ATTENZIONE:

- adottare quanto più possibile pattern progettuali che permettono di risolvere problemi già noti. Nel caso di adozione di pattern vanno indicati: il pattern adottato e il problema che è stato risolto attraverso il pattern. Vanno introdotti poi le classi nella parte strutturale dell'architettura e adattate le altre viste
- adottare quanto più possibile i Design Principles
- seguire le indicazioni per la progettazione "sicura"
- far emergere le strutture dati che erano implicite nell'analisi (liste, mappe, etc.). Il progettista ha l'obbligo di creare diagrammi che siano direttamente implementabili, non lasciate troppe scelte all'implementatore, potrebbe involontariamente inserire buchi nella sicurezza.

L'artefatto da produrre al termine di questa fase è l'**Architettura del Sistema**. Come per l'architettura logica dovranno essere mostrate le tre diverse viste.

Le diverse sezioni da considerare sono le seguenti.

PROGETTAZIONE ARCHITETTURALE

Requisiti non funzionali

Si analizzano i requisiti non funzionali definiti nelle tabelle dell'analisi e si fa una valutazione dei trade-off tra i requisiti

Scelta dell'architettura

Si sceglie l'architettura (o diverse architetture per parti diverse del sistema) alla luce dei trade-off tra i requisiti tenendo conto anche dei diversi pattern architetturali più conosciuti.

Scelte tecnologiche (opzionale)

Questa sezione è da compilare solo nel caso si intenda scegliere le tecnologie (intese anche come linguaggio o linguaggi di programmazione) con cui sarà implementato il sistema già in fase di progettazione. Se, invece, si demanda la scelta alla successiva fase di implementazione, questa parte va trascurata.

PROGETTAZIONE DI DETTAGLIO

Struttura

Come per l'architettura logica riportare prima un diagramma che mostra l'architettura generale organizzata in package o componenti e successivamente i diagrammi delle classi (partire da quelli dell'analisi, raffinarli e aggiungere nuove classi di progettazione) di dettaglio per ogni package. Importante: le classi devono essere quelle che poi vanno implementate, quindi devono avere tutti metodi e gli attributi necessari, le navigabilità e cardinalità devono essere tutte specificate.

Specificare bene quali pattern sono stati adottati e quali design principle sono stati seguiti.

Interazione

Progettare nel dettaglio i protocolli di interazione tra le classi già esistenti e le classi introdotte nelle fase di progettazione

Comportamento

Progettare nel dettaglio lo stato interno delle classi principali → raffinamento dei diagrammi di stato dell'analisi e introduzione dei diagrammi di stato per le nuove classi introdotte

PROGETTAZIONE DELLA PERSISTENZA

Indicare l'eventuale diagramma E-R per il database e/o il formato di tutti i file utilizzati

PROGETTAZIONE DEL COLLAUDO

Indicare esempi NUnit/JUnit del collaudo delle principali classi dell'application logic

PROGETTAZIONE DEL DEPLOYMENT

Questa sezione va organizzata in due parti

Progettazione del deployment per la sicurezza

Seguire tutte le indicazioni per progettare il deployment per la sicurezza

Deployment del sistema

Qua va riportato il diagramma degli artefatti e il diagramma del deployment logico

Implementazione

Scelte tecnologiche (opzionale)

In questa sezione si dovranno discutere i motivi che hanno portato alla scelta delle tecnologie (ivi compresi anche il linguaggio o i linguaggi scelti per l'implementare il prototipo) che saranno usate nell'implementazione. È opzionale perché se le scelte sono state fatte in fase di progettazione saranno già state discusse, mentre se il progetto è stato lasciato neutro qua si dovranno discutere le motivazioni che hanno portato a scegliere una tecnologia piuttosto che un'altra.

Scelte Implementative

Qua vanno indicate le modifiche al progetto che sono dovute:

- a. scelta della tecnologie
- b. per "mancanze" nella progettazione (servivano metodi in più / aggiunta strutture dati etc.)

Se occorre riportare anche i diagrammi UML variati rispetto alla progettazione

Collaudo

Valutazione della fase di collaudo dei singoli componenti e dei test di integrazione

Deployment (opzionale)

Eventuale diagramma di deployment fisico nel quale indicare su quali nodi fisici saranno allocate le diverse parti del sistema. Indicare eventuali nodi per la replicazione del servizio.