

TP



Documento di Test Planning

Gruppo 14

Team Manager: Maria D'Arco 0510201593

Componenti :

Massimo Caruso	0510201704
Filomena Fruncillo	0510201659
Vincenzo D'Alessio	0510201560
Giuseppe Valitutto	0510201191

1.	Introduzione.....	3
2.	Documenti Correlati	4
2.1	Relazioni con il Documento di Raccolta e Analisi dei requisiti (RAD).....	4
2.2	Relazioni con il System Design Document (SDD).....	4
3.	Overview del sistema.....	5
4.	Funzionalità testate.....	6
5.	Criteri Pass/Failed.....	7
6.	Approccio.....	7
6.1	Testing d'Unità	7
6.2	Testing d'Integrazione.....	8
6.3	Testing di Sistema.....	8
7.	Sospensione e Ripresa.....	8
7.1	Criteri di sospensione	8
7.2	Criteri di ripresa.....	9
8.	Materiale per il Testing (Requisiti Hardware/software).....	9
9.	Casi di Test (Test Cases).....	10
9.1	Login.....	10
9.2	Gestione Calendario.....	12
9.2.1	Visualizza appuntamenti	15
9.3	Gestione Personale.....	17
9.3.1	Modifica Personale/Ricerca Personale.....	17
9.3.2	Inserimento Personale.....	21
9.4	Gestione Sale.....	32
9.4.1	Inserimento Sala	32
9.4.2	Ricerca Sala/Modifica Sala.....	34
9.5	Gestione Attrezzature.....	36
9.5.1	Inserimento Attrezzatura	36
9.5.2	Modifica Attrezzatura/Ricerca Attrezzatura.....	41
10	Pianificazione del Testing.....	43
10.1	Determinazione dei Ruoli.....	44
10.2	Determinazione dei Rischi.....	44
10.3	Decomposizione gerarchia del sistema.....	44
10.4	Organizzazione delle attività di Testing.....	45
10.5	Schedulazione delle attività di Testing	46
11.	Glossario.....	47







1.Introduzione

La *Fault Detection* è una delle norme adottate per aumentare l'affidabilità di un sistema software. Essa consiste nel uso tecniche, come quella di *Testing*, per rilevare , in maniera pianificata, gli errori all'interno del codice prodotto dal mapping.

L'obiettivo del Testing è quello di evitare che tali errori si presentino in fase di utilizzo del sistema da parte dell'utente finale.



Nel presente documento vengono descritte e organizzate le attività di Testing, relative alle funzionalità implementate e specificate nell'ODD . In particolare ogni funzionalità verrà testata dal punto di vista degli utenti che possono far uso di essa.

Nello specifico le funzionalità prese in esame sono relative alla:

-  Gestione degli Accessi;
-  Gestione del Personale;
-  Gestione delle Attrezzature;
-  Gestione Fornitori;
-  Gestione delle Sale;
-  Gestione del Calendario.

2. Documenti correlati

Innanzitutto è da precisare che il Test Plan ha una relazione notevole con gli altri documenti precedentemente prodotti.

-  In prima luogo ha un forte legame con la fase di mapping oggetti/codice e con l'object Design Document (ODD).
-  In secondo luogo si specifica che si è deciso di passare alla fase di Testing dopo che la fase d'impelementazione fosse ben avviata, in modo tale da osservare e rilevare il comportamento dell'esecuzione del software rispetto a quello atteso e documentato nei System Models.

2.1 Relazioni con il documento di analisi dei requisiti (RAD)

Le specifiche, quali requisiti funzionali e non funzionali, espresse nel documento di analisi dei requisiti sono di particolare riferimento del test delle funzionalità.




2.2 Relazioni con il System Design Document (SDD)

I vari test sulle componenti dovranno mantenersi fedeli alla decomposizione in sottosistemi specificati nel documento di System Design (SDD).

Il test d'integrazione farà riferimento ai package definiti nel documento di Object Design (ODD).

3 Overview del Sistema

Il sistema verrà suddiviso in tre sottosistemi, precedentemente specificati nell'SDD, quali:

-  Sottosistema Amministrazione;
-  Sottosistema Segreteria;
-  Sottosistema Medico.

In ognuno di essi le componenti sono suddivise in base ai diversi tipi di gestione, di seguito riportate:

Sottosistema Amministrazione:

- Gestione Sale;
- Gestione Personale;
- Gestione Attrezzature;
- Gestione Fornitori;
- Gestione Statistiche.






Sottosistema Segreteria:

- Gestione Calendario:
 - Gestione Appuntamenti;
 - Gestione Richiami Periodici;
 - Stampa Moduli.

Sottosistema Medico:

- Gestione Cartelle Cliniche;
- Gestione Dati Paziente;
- Richieste Amministrative.

Come si evince dall'ODD non tutte le funzionalità sono state implementate ma sono state realizzate solo quelle facenti parte dei sottosistemi scelti:

-  GestioneAccessi;
-  GestioneSale;
-  GestionePersonale;
-  GestioneAttrezzature;
-  GestioneFornitori;

 GestioneCalendario (ma non viene implementata la stampa moduli)

4 Funzionalità testate

Tra le funzionalità implementate verranno testate:

Gestione Accessi:

- ✓ Login

Gestione Personale:

- ✓ Inserimento Personale;
- ✓ Ricerca Personale;
- ✓ Modifica Peronale;

Gestione Sale:

- ✓ Inserimento Sala;
- ✓ Ricerca Sala;
- ✓ Modifica Sala;

Gestione Attrezzature:

- ✓ Inserimeto Attrezzature;
- ✓ Aggiorna Attrezzatura;
- ✓ Ricerca Attrezzatura;
- ✓ Ordina Attrezzatura;
- ✓ Modifica Attrezzatua;

Gestione Fornitori:

- ✓ Inserimento Fornitore;

Gestione Calendario:

- ✓ Inserisci Appuntamento;
- ✓ Ricerca Appuntamento;
- ✓ Visualizza Calendario;
- ✓ Modifica Appuntamento.

5 Criteri Pass/Failed

Nei paragrafi successivi verranno definiti i criteri per stabilire quando un dato input avrà superato o meno un test.

I dati in input verranno classificati in classi d'equivalenza, ovvero raggruppati in insiemi dalle caratteristiche comuni, per i quali sarà sufficiente testare un solo elemento rappresentativo dell'insieme d'appartenenza (test case - WECT).

Superano il test i dati input che risponderanno alle specifiche output definite dall'Oracolo del test case, ovvero lo sviluppatore o membro del team di testing che conosce il risultato corretto che il sistema deve presentare. In tal caso l'intera classe di appartenenza verificherà la correttezza del test, in caso contrario, l'intera classe non supererà il test se non risponde alle aspettative di output previste.

6 Approccio

Per testare in dettaglio ciascuna unità, le tecniche di testing inizialmente adottate riguarderanno il Testing di Unità dei singoli componenti.

Successivamente verrà adoperato il Testing d'Integrazione per evidenziare il test delle interface sulle unità interessate.

Infine si attuerà il Testing di Sistema, che esaminerà l'intero sistema assemblato nei suoi componenti, ma soprattutto servirà per verificare che il sistema soddisfi le richieste del committente.

6.2 Testing di Unità

In questa fase verranno ricercate le condizioni di fallimento isolando, i componenti del sistema, attraverso l'utilizzo di test driver e test stub, ovvero attraverso l'utilizzo di implementazioni parziali di componenti che dipendono o da cui dipendono le componenti in esame.

Per tale Testing, dato il ristretto tempo a disposizione, verrà utilizzata la strategia basata esclusivamente sulla tecnica della *BlackBox* . Tale tecnica permette di evidenziare il comportamento Input/Output ignorando la struttura dell'intero componente. Gli errori o i fallimenti del sistema, individuati in questa fase, dovranno

essere comunicati opportunamente agli implementatori al fine di apportare le dovute modifiche al sistema per poi ripristinare al più presto il Testing.

Non viene riportata nel presente documento la documentazione relativa alla *WhiteBox Testing* in quanto la sua realizzazione è stata effettuata durante la fase implementativa in modo totalmente implicito da parte degli sviluppatori.

6.3 Testing d'integrazione

Verranno testate nel complesso le componenti di una funzionalità, attraverso l'utilizzo di una strategia Sandwich (derivante dalla combinazione delle tecniche Bottom-Up e Top-Down) .

Si procederà, poi, integrando a funzionalità successive fino ad esaurire le funzionalità implementate. In questo modo si mira a ridurre le dipendenze tra le differenti funzionalità e a facilitare la ricerca di errori nella comunicazione tra le interfacce e i sottosistemi.

6.4 Testing di Sistema

Lo scopo della fase di Testing è quello di dimostrare che il sistema soddisfi i requisiti richiesti e sia, quindi, pronto all'uso.

Come per i Testing d'Unità si cercherà di testare sia le funzionalità più importanti per l'utente, sia quelle che hanno una maggiore probabilità di fallimento. Inoltre anche in questa fase si adopererà la tecnica della *BlackBox*.

7 Sospensione e Ripresa

7.2 Criteri di sospensione

Quando saranno testate tutte le classi per ogni possibile input e il risultato ottenuto sarà quello atteso, allora, solo in questo caso il Testing sarà sospeso.

Nel caso in cui durante la fase di Testing sarà individuato un errore, si dovrà passare alla fase di correzione, che sarà seguita da una successiva ripetizione dell'intero Testing per un'ulteriore verifica di errori creati anche dall'introduzione della correzione stessa.

7.3 Criteri di ripristino

Come già accennato in precedenza una ripresa del Testing sarà eseguita dopo la fase di correzione di un errore, in modo da individuare la presenza di nuovi errori immessi anche dall'introduzione della correzione stessa.

8 Materiale per il testing (Requisiti Hardware/software)

L'hardware necessario per l'attività di test è un PC su cui sia installato il DBMS MySQL 5.0 e una Java Virtual Machine in grado di supportare Java 6.

9 Casi di test (Test Cases)

9.1 Login

La funzionalità da testare riguarda il componente GestioneAccessi, in particolare la funzionalità di Login, cioè di accesso al sistema.

Classi di equivalenza

Gli input del test sono l'userId e la password dell'utente.

INPUT	userId,password	
Classi Valide	C_001	Stringa alfanumerica compresa tra 4 e 10 caratteri.
Classi non valide	C_002	Stringa alfanumerica strettamente maggiore di 10 di caratteri.
	C_003	Stringa alfanumerica strettamente minore di 4 caratteri.
	C_004	Stringa passata non valida o nulla.

Identificazione Test Case:

Test case: TCLogin_01		
Input	Classe	Membro
userId	C_002	Giuseppe_Valitutto7
password	C_001	04_02_86

Test case: TCLogin_02		
Input	Classe	Membro
userId	C_001	Massimo86
password	C_002	Massimo_caruso

Test case: TCLogin_03		
Input	Classe	Membro
userId	C_001	Admin
password	C_001	Admin

Test case: TCLogin_04		
Input	Classe	Membro
userId	C_002	Filomena_Fruncillo
password	C_002	Fruncillo88

Test case: TCLogin_05		
Input	Classe	Membro
userId	C_003	Enz
password	C_003	Dal

Test case: TCLogin_06		
Input	Classe	Membro
userId	C_004	Stringa passata null
password	C_004	Stringa passata null

9.2 Gestione Calendario

Il test riguarda la componente GestioneCalendario in particolare la funzionalità d' inserimento di un Appuntamento.

Classi di equivalenza

Gli input sono: nomePaziente, cognomePaziente, nomeMedico, cognomeMedico, tipoServizio,richiamo,dataAppuntamento,ora.

Input	nomePaziente,cognomePaziente	
Classi Valide	C_005	Stringa
Classi non valide	C_006	Viene passata un stringa nulla.

Input	nomeMedico,cognomeMedico	
Classi Valide	C_007	Stringa
Classi non valide	C_008	Viene passata un stringa nulla.

Input	tipoServizio	
Classi Valide	C_009	Stringa
Classi non valide	C_010	Viene passata un stringa nulla.

Input	dataAppuntamento	
Classi Valide	C_013	Oggetto di tipo Data in formato GG/MM/AAAA
Classi non valide	C_014	Formato del tipo Data errato
	C_015	Data non inserita

Input	Ora	
Classi Valide	C_016	Attributo di tipo Data con formato ora,minuti.
Classi non valide	C_017	Formato di tipo Data errato.
	C_018	Data non inserita

Identificazione Test Case

Test Case :TCInserimentoDatiAppuntamento_01		
Input	Classe	Membro
nomePaziente	C_005	Massimo
cognomePaziente	C_005	Caruso
nomeMedico	C_007	Vincenzo
cognomeMedico	C_007	D'Alessio
tipoServizio	C_009	Igienista
dataAppuntamento	C_013	28/06/09
ora	C_016	11:00

Test Case :TCInserimentoDatiAppuntamento_02		
Input	Classe	Membro
nomePaziente	C_005	Giuseppe
cognomePaziente	C_005	Rossi
nomeMedico	C_007	Antonio

cognomeMedico	C_007	Cupo
tipoServizio	C_010	Stringa nulla
dataAppuntamento	C_013	28/06/09
ora	C_016	11:00

Test Case :TCInserimentoDatiAppuntamento_03

Input	Classe	Membro
nomePaziente	C_005	Filomena
cognomePaziente	C_005	Fruncillo
nomeMedico	C_007	Vincenzo
cognomeMedico	C_007	D'Alessio
tipoServizio	C_009	Igienista
dataAppuntamento	C_014	28giugno2009
ora	C_016	12:00

Test Case :TCInserimentoDatiAppuntamento_04

Input	Classe	Membro
nomePaziente	C_006	Stringa nulla
cognomePaziente	C_006	Stringa nulla
nomeMedico	C_007	Pietro
cognomeMedico	C_007	Caporale
tipoServizio	C_009	implantologia
dataAppuntamento	C_013	29/06/09
ora	C_016	12:00

Test Case :TCInserimentoDatiAppuntamento_05

Input	Classe	Membro
nomePaziente	C_005	Maddalena
cognomePaziente	C_005	Vitolo
nomeMedico	C_008	Stringa nulla
cognomeMedico	C_008	Stringa nulla
tipoServizio	C_009	parodontologia
dataAppuntamento	C_013	29/06/09
ora	C_016	12:00

Test Case :TCInserimentoDatiAppuntamento_06

Input	Classe	Membro
nomePaziente	C_006	Stringa nulla

cognomePaziente	C_006	Stringa nulla
nomeMedico	C_008	Stringa nulla
cognomeMedico	C_008	Stringa nulla
tipoServizio	C_010	Stringa nulla
dataAppuntamento	C_013	29/05/09
ora	C_018	Data non inserita

9.2.1 Visualizza appuntamenti

Il test riguarda la componente Gestione Calendario in particolare la funzionalità di visualizzazione degli appuntamenti.

Classi di equivalenza

Gli input sono: dataAppuntamento e tipoServizio.

Input	data Appuntamento	
Classi valide	C_019	oggetto di tipo data, con formato GG/MM/AAAA
Classi non valide	C_020	data non inserita

Input	tipoServizio	
Classi valide	C_021	stringa
Classi non valide	C_022	stringa non inserita

Identificazione Test Case

Test Case: TCVisualizzaAppuntamento_01		
Input	Classe	Membro
dataAppuntamento	C_019	17-mag-2009
tipoServizio	C_021	parodontologia

Test Case:TCVisualizzaAppuntamento_02		
Input	Classe	Membro
dataAppuntamento	C_019	9-mag-2009
tipoServizio	C_022	stringa nulla

Test Case:TCVisualizzaAppuntamento_03		
Input	Classe	Membro
dataAppuntamento	C_020	stringa nulla
tipoServizio	C_022	stringa nulla

Test Case:TCVisualizzaAppuntamento_04		
Input	Classe	Membro
dataAppuntamento	C_020	stringa nulla
tipoServizio	C_021	implantologia

9.2.2 RicercaAppuntamento

Il test riguarda la componente GestioneCalendario in particolare la funzionalità di ricerca appuntamento per ora.

Classi di equivalenza

L'input è l' ora dell'appuntamento.

Input	Ora	
Classi Valide	C_023	Attributo di tipo Data con formato ora,minuti.
Classi non valide	C_024	Formato di tipo Data errato.

Identificazione Test Case

Test Case:TCRicercaAppuntamento_01		
Input	Classe	Membro
ora	C_023	11:00

Test Case:TCRicercaAppuntamento_02		
Input	Classe	Membro
ora	C_024	11

9.3 Gestione Personale

9.3.1 ModificaPersonale/RicercaPersonale

Il test riguarda il componente Gestione Personale, in particolare la funzionalità di ModificaPersonale.

La modifica del personale si divide in due parti: RicercaPersonale, che verifica l'effettiva presenza del dipendente nel database, e ModificaPersonale, per l'effettiva modifica.

Classi di equivalenza

L'input è il codfisc.

Input	codfisc	
Classi valide	C_025	stringa alfanumerica di 16 caratteri
Classi non valide	C_026	stringa alfanumerica di lunghezza minore di 16 caratteri
	C_027	stringa alfanumerica di lunghezza maggiore di 16 caratteri
	C_028	stringa nulla

Identificazione Test Case

Test Case:TCRicercaPersonale_01		
Input	Classe	Membro
codfisc	C_025	dlsvcn83e25a717u

Test Case:TCRicercaPersonale_02		
Input	Classe	Membro
codfisc	C_026	dlsvcn83e25a71

Test Case:TCRicercaPersonale_03		
Input	Classe	Membro
codfisc	C_027	dlsvcn83e25a717urt

Test Case:TCRicercaPersonale_04		
Input	Classe	Membro
codfisc	C_028	Stringa nulla

Dopo la fase di ricerca è possibile, se il paziente è stato trovato , modificare i seguenti attributi:residenza,tel, tipo, salario e data.

Classi di equivalenza

L'input è : residenza,tel,tipo,salario,data.

Input	residenza	
Classi valide	C_029	stringa
Classi non valide	C_030	stringa non inserita

Input	Tel	
Classi valide	C_031	Intero di lunghezza minore o uguale a 10.
Classi non valide	C_032	Intero di lunghezza maggiore di 10.
	C_033	Intero non inserito.

Input	Tipo	
Classi valide	C_034	stringa
Classi non valide	C_035	stringa non inserita

Input	Salario	
Classi valide	C_036	intero
Classi non valide	C_037	intero non inserito

Input	Data	
Classi valide	C_038	oggetto di tipo data, con formato GG/MM/AAAA
Classi non valide	C_039	Formato della data errato
	C_040	Data non inserita

Identificazione Test Case

Test Case: TCModificaPersonale_01		
Input	Classe	Membro
residenza	C_029	Avellino
tel	C_031	0825761976
tipo	C_034	Indeterminato
salario	C_036	500
data	C_038	12/03/1988

Test Case: TCModificaPersonale_02		
Input	Classe	Membro
residenza	C_030	Stringa nulla
tel	C_031	0825761976
tipo	C_034	Indeterminato
salario	C_036	500
data	C_038	12/03/1988

Test Case: TCModificaPersonale_03		
Input	Classe	Membro
residenza	C_029	Avellino
tel	C_032	0825761976287648912345
tipo	C_034	Indeterminato
salario	C_035	500
data	C_038	12/03/1988

Test Case:TCModificaPersonale_04		
Input	Classe	Membro
residenza	C_029	Avellino
tel	C_031	0825761976
tipo	C_035	Stringa nulla
salario	C_036	500
data	C_038	12/03/1988

Test Case:TCModificaPersonale_05		
Input	Classe	Membro
residenza	C_029	Avellino
tel	C_033	Intero nullo
tipo	C_034	parttime
salario	C_037	Intero nullo
data	C_038	12/03/1988

Test Case:TCModificaPersonale_06		
Input	Classe	Membro
residenza	C_029	Avellino
tel	C_033	Intero nullo
tipo	C_034	parttime
salario	C_036	500
data	C_039	1299/03/1988

Test Case:TCModificaPersonale_07		
Input	Classe	Membro
residenza	C_030	stringa nulla
tel	C_033	Intero nullo
tipo	C_035	stringa nulla
salario	C_037	Intero nullo
data	C_040	1299/03/1988

9.3.2 Inserimento Personale

Il test riguarda la componente Gestione Personale in particolare la funzionalità di inserimento del personale.

Classi di equivalenza

Gli input

sono:codfisc,nome,cognome,residenza,dataNascita,luogoNascita,tel,tiposesso,tiposalario,data.

Input	codfisc	
Classi valide	C_041	stringa alfanumerica di 16 caratteri
Classi non valide	C_042	stringa alfanumerica di lunghezza minore di 16 caratteri
	C_043	stringa alfanumerica di lunghezza maggiore di 16 caratteri
	C_044	stringa nulla

Input	nome,cognome,residenza,luogo Nascita,tip	
Classi valide	C_045	Stringa
Classi non valide	C_046	stringa non inserita

Input	Tel	
Classi valide	C_047	Intero di lunghezza minore o uguale a 20.
Classi non valide	C_048	Intero di lunghezza maggiore di 20
	C_049	Intero non inserito.

Input	data Nascita	
Classi valide	C_050	oggetto di tipo data,con formato GG/MM/AAAA
Classi non valide	C_051	Formato della data inserito in modo errato
	C_052	data non inserita

Input	Sesso	
Classi valide	C_053	Carattere f/m
Classi non valide	C_054	Stringa diversa da f/m
	C_055	stringa non inserita

Input	Tipo	
Classi valide	C_056	Stringa
Classi non valide	C_057	stringa non inserita

Input	Salario	
Classi valide	C_058	Intero
Classi non valide	C_059	intero non inserito

Input	Data	
Classi valide	C_060	oggetto di tipo data, con formato GG/MM/AAAA
Classi non valide	C_061	Formato inserito in modo errato
	C_062	data non inserita

Identificazione Test Case

Test Case: TCInserimentoPersonale_01		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	10/09/1988
luogoNascita	C_045	Avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_02		
Input	Classe	Membro
codfisc	C_043	dlsvcn83e25a717utyu
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	10/09/1988
luogoNascita	C_045	Avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case:TCInserimentoPersonale_03		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_046	stringa nulla
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	Avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case:TCInserimentoPersonale_04		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_046	Stringa nulla
dataNascita	C_050	19/09/1988
luogoNascita	C_045	Avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case:TCInserimentoPersonale_05		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_051	1988/09/19
luogoNascita	C_045	Avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
seesso	C_053	f
tipo	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case:TCInserimentoPersonale_06		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_046	Stringa nulla
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
seesso	C_053	f
tipo	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_07		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_046	Stringa nulla
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_08		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_045	forino
tel	C_048	08257619762356
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_09		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_046	Stringa nulla
seesso	C_053	f
tipo	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_10		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
seesso	C_054	femmina
tipo	C_056	Indeterminato
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_11		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_057	Stringa nulla
salario	C_058	500
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_12		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
sex	C_053	f
tip	C_056	Indeterminato
salario	C_059	Valore nullo
data	C_060	12/05/2008

Test Case: TCInserimentoPersonale_13		
Input	Classe	Membro
codfisc	C_041	dlsvcn83e25a717u
nome	C_045	filomena
cognome	C_045	fruncillo
dataNascita	C_050	19/09/1988
luogoNascita	C_045	avellino
residenza	C_045	forino
tel	C_047	0825761976
tipo	C_045	Segretaria
seesso	C_053	f
tipo	C_056	Indeterminato
salario	C_058	500
data	C_062	Data non inserita

Test Case: TCInserimentoPersonale_14		
Input	Classe	Membro
codfisc	C_043	dlsvcn83e25a717uerty
nome	C_046	stringa nulla
cognome	C_046	stringa nulla
dataNascita	C_051	1988/09/19
luogoNascita	C_046	stringa nulla
residenza	C_046	stringa nulla
tel	C_048	08257619762356
tipo	C_046	stringa nulla
seesso	C_054	femmina
tipo	C_057	stringa nulla
salario	C_059	Intero non inserito
data	C_062	Data non inserita

9.4 Gestione Sale

9.4.1 Inserimento Sala

Il test riguarda la componente Gestione Sala in particolare la funzionalità inserimento DatiSala.

Classi di equivalenza

Gli input sono: codice Sala, nomeSala, descrizione, codice Fiscale.

Input	codice Sala	
Classi Valide	C_063	Valore di tipo int
Classi non Valide	C_064	Valore diverso da int
	C_065	Valore nullo

Input	nomeSala	
Classi Valide	C_066	Stringa compresa tra 3 e 20 caratteri
Classi non Valide	C_067	Stringa maggiore di 20 caratteri
	C_068	Stringa minore di 3 di caratteri
	C_069	Stringa nulla

Input	Descrizione	
Classi Valide	C_070	Stringa di 45 caratteri
Classi non Valide	C_071	Stringa maggiore di 45 caratteri
	C_072	Stringa nulla

Identificazione TestCase

Test Case: TCInserimentoDatiSala_01		
Input	Classe	Membro
codiceSala	C_063	111
nomeSala	C_066	Igienista
descrizione	C_070	Pulizia denti

Test Case: TCInserimentoDatiSala_02		
Input	Classe	Membro
codiceSala	C_064	11.21
nomeSala	C_067	Giuseppegiuseppegiuseppe
descrizione	C_072	Stringa nulla

Test Case: TCInserimentoDatiSala_03		
Input	Classe	Membro
codiceSala	C_065	valore nullo
nomeSala	C_069	Stringa nulla
descrizione	C_072	Stringa nulla

Test Case: TCInserimentoDatiSala_04		
Input	Classe	Membro
codiceSala	C_063	112
nomeSala	C_066	Paradontologia
descrizione	C_070	aaaaaaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaaaaaa

Test Case: TCInserimentoDatiSala_05		
Input	Classe	Membro
codiceSala	C_063	113
nomeSala	C_066	Stringa nulla
descrizione	C_070	Ricostruzione denti

9.4.2 RicercaSala/ModificaSala

Il test riguarda il componente GestioneSala, in particolare la funzionalità di ModificaSala.

La modifica della Sala si divide in due parti: RicercaSala, che verifica l'effettiva presenza di una sala nel database, e ModificaSala, per l'effettiva modifica.

Classi di equivalenza

Input	codiceSala	
Classi Valide	C_073	Valore di tipo int
Classi non valide	C_074	Valore diverso da int
	C_075	Valore nullo

Identificazione Test Case

TestCase:TCRicercaSala_01		
Input	Classe	Membro
codiceSala	C_073	111

TestCase:TCRicercaSala_02		
Input	Classe	Membro
codiceSala	C_074	11.21

TestCase:TCRicercaSala_03		
Input	Classe	Membro
codiceSala	C_075	Valore nullo

Test Case: TCRicercaSala_04		
Input	Classe	Membro
codiceSala	C_073	113

Dopo la fase di ricerca è possibile, se la sala è stata trovata , modificare i seguenti attributi: nomeSala,descrizione.

Input	nomeSala	
Classi Valide	C_076	Stringa compresa tra 3 e 20 caratteri
Classi non Valide	C_077	Stringa maggiore di 20 caratteri
	C_078	Stringa minore di 3 di caratteri
	C_079	Stringa nulla

Input	Descrizione	
Classi Valide	C_080	Stringa di 45 caratteri
Classi non Valide	C_081	Stringa maggiore di 45 caratteri
	C_082	Stringa nulla

Identificazione TestCase

Test Case: TCModificaDatiSala_01		
Input	Classe	Membro
nomeSala	C_076	Igienista
descrizione	C_080	Pulizia denti

Test Case: TCModificaDatiSala_02		
Input	Classe	Membro
nomeSala	C_076	paradontologia
descrizione	C_081	aaaaaaaaaaaaaaaaaaaaaaaaa aaaaaaaaaaaaaaaaaaaaaaaaa

Test Case: TCModificaDatiSala_03		
Input	Classe	Membro
nomeSala	C_079	Stringa nulla
descrizione	C_080	Ricostruzione denti

9.5 Gestione Attrezzature

9.5.1 InserimentoAttrezzatura

Il test riguarda la componente Gestione Attrezzature in particolare la funzionalità di inserimento delle attrezzature.

Classi di equivalenza

Gli input sono del test sono: codiceTipo, nome, quantità, costo, tipo, descrizione, locazione, codiceFornitore.

Input	codice Tipo	
Classi valide	C_083	Intero
Classi non valide	C_084	Intero non inserito

Input	nome,tipo,descrizione,locazione	
Classi valide	C_085	Stringa
Classi non valide	C_086	stringa non inserita

Input	quantità	
Classi valide	C_087	Intero
Classi non valide	C_088	Intero negativo
	C_089	Intero non inserito.

Input	codiceFornitore	
Classi valide	C_090	Stringa di lunghezza uguale a 16
Classi non valide	C_091	Stringa di lunghezza minore di 16
	C_092	Stringa di lunghezza maggiore di 16
	C_093	Stringa non inserita

Input	costo	
Classi valide	C_091	Double inserito
Classi non valide	C_092	Attributo non valido

Identificazione Test Case

Test Case: TCInserimentoAttrezzatura_01		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_087	3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_085	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case: TCInserimentoAttrezzatura_02		
Input	Classe	Membro
codiceTipo	C_084	Intero non inserito
nome	C_086	Stringa non inserita
quantità	C_088	-3
costo	C_092	X
tipo	C_086	Stringa non inserita
descrizione	C_086	Siringa non inserita
locazione	C_086	Stringa non inserita
codiceFornitore	C_091	Wer234a

Test Case: TCInserimentoAttrezzatura_03		
Input	Classe	Membro
codiceTipo	C_084	Intero non inserito
nome	C_085	Siringa
quantità	C_087	3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_086	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case:TCInserimentoAttrezzatura_04		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_086	Stringa non inserita
quantità	C_087	3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_085	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case:TCInserimentoAttrezzatura_05		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_088	-3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_085	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case:TCInserimentoAttrezzatura_06		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_087	3
costo	C_092	X
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_085	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case:TCInserimentoAttrezzatura_07		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_087	3
costo	C_091	10.68
tipo	C_086	Stringa non inserita
descrizione	C_085	Siringa del tipo vdt
locazione	C_085	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case:TCInserimentoAttrezzatura_08		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_087	3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_086	Stringa non inserita
locazione	C_085	A01
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case:TCInserimentoAttrezzatura_09		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_087	3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_086	Stringa non inserita
codiceFornitore	C_090	Wer234asd34dsf4t

Test Case: TCInserimentoAttrezzatura_10		
Input	Classe	Membro
codiceTipo	C_083	2345543
nome	C_085	Siringa
quantità	C_087	3
costo	C_091	10.68
tipo	C_085	Attrezzature
descrizione	C_085	Siringa del tipo vdt
locazione	C_085	A01
codiceFornitore	C_091	Wer234a

9.5.2 ModificaAttrezzatura/RicercaAttrezzature

Il test riguarda il componente GestioneAttrezzature, in particolare la funzionalità di ModificaAttrezzature.

La modifica dei dati di un'attrezzatura si divide in due parti: RicercaAttrezzature, che verifica l'effettiva presenza dell'attrezzatura nel database, e ModificaAttrezzature, per l'effettiva modifica.

Classi di equivalenza

Gli input sono: codiceTipo

Input	codice Tipo	
Classi valide	C_094	Intero
Classi non valide	C_095	Intero non inserito

Identificazione Test Case

Test Case:TCRicercaAttrezzatura_01		
Input	Classe	Membro
codiceTipo	C_094	12345

Test Case:TCRicercaAttrezzatura_02		
Input	Classe	Membro
codice Tipo	C_095	Intero non inserito

Dopo la fase di ricerca è possibile, se l'attrezzatura è stata trovata , modificare i seguenti attributi: locazione,descrizione.

Classi di equivalenza

Input	locazione	
Classi valide	C_096	Stringa
Classi non valide	C_097	stringa non inserita

Input	locazione	
Classi valide	C_098	Stringa
Classi non valide	C_099	Stringa superiore a 150 caratteri

Identificazione Test Case

Test Case:TCModificaDatiAttrezzatura_01		
Input	Classe	Membro
descrizione	C_098	Siringa del tipo vdt

locazione	C_096	A71
-----------	-------	-----

Test Case:TCModificaDatiAttrezzatura_02		
Input	Classe	Membro
descrizione	C_099	Stringa superiori a 150 caratteri
locazione	C_097	Stringa non inserita

Test Case:TCModificaDatiAttrezzatura_03		
Input	Classe	Membro
locazione	C_096	A01
descrizione	C_099	Stringa superiore a 150 caratteri

Test Case:TCModificaDatiAttrezzatura_04		
Input	Classe	Membro
locazione	C_097	Stringa non inserita
descrizione	C_098	Materiali d'uso

10 Pianificazione del Testing

Il Testing viene eseguito dalle parti del team che hanno buona conoscenza del sistema e delle tecniche di Testing da adottare, in rispetto dei vincoli di tempo, qualità e budget precedentemente stabiliti.

Il team avrà a disposizione non più di tre giorni e non ci saranno limitazioni a livello di budget, in quanto il sistema è sviluppato prettamente per scopi didattici. Rispetto alla qualità, invece, si cercherà di scovare il più alto numero possibile d'errori, sempre nel rispetto dei limiti di tempo stabiliti.

Normalmente i componenti del team di Testing sono persone che non siano state coinvolte nella fase di sviluppo, in quanto è statisticamente provato che uno sviluppatore è molto restio a modificare il proprio codice. Nel nostro caso specifico tale approccio non potrà essere pienamente rispettato in quanto, dati i fini didattici del progetto, si è preferito rendere partecipe ad ogni fase di sviluppo tutti i componenti del team il cui numero dei componenti sarebbe stato insufficiente per compiere interamente tutti i task di sviluppo. Per la documentazione di eventuali *fault* di una

componente verranno notificati agli sviluppatori della componente stessa, per consentire la correzione del sistema. Inseguito alle correzioni e alle revisioni, il sistema verrà nuovamente testato per verificare se le modifiche apportate per risolvere gli errori individuati non ne abbiano introdotti nuovi.

10.2 Determinazione dei ruoli

L'attività di Testing di sistema sarà svolta, come detto, dai componenti del team che per primi completeranno l'implementazione. Sarà compito del team manager quello di coordinare e revisionare il lavoro.

10.3 Determinazione dei rischi

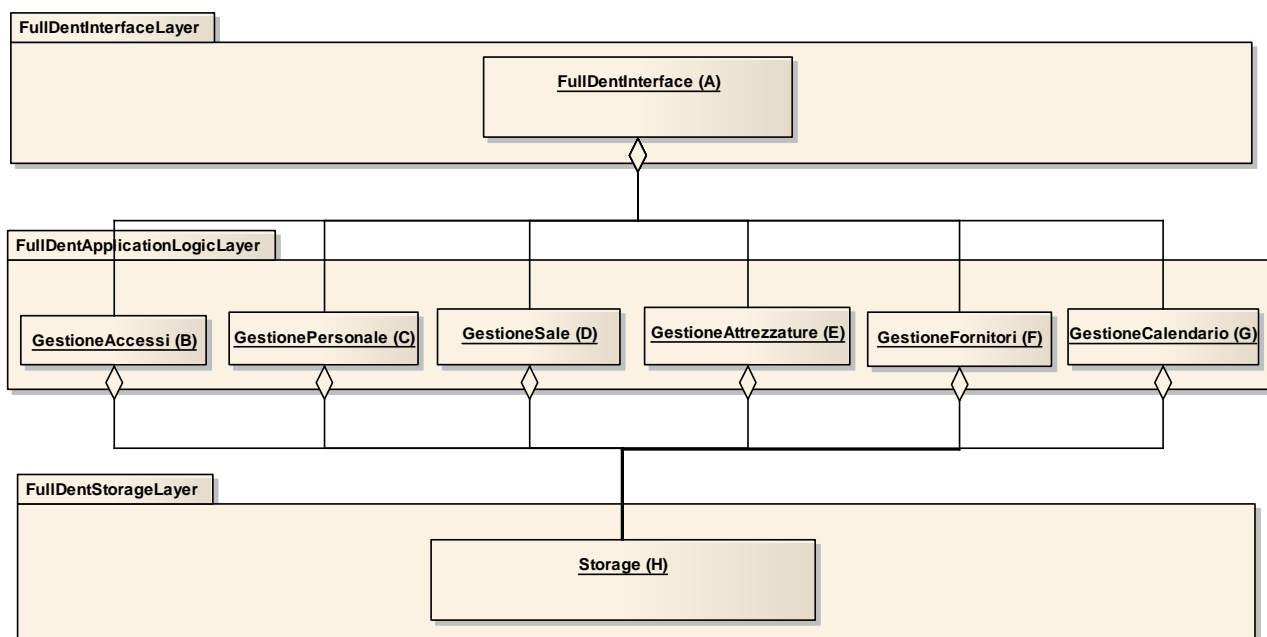
Con una buona pianificazione del Testing si cercherà di minimizzare i rischi di fallimento. In caso di ritardi dovuti all'individuazione di un numero troppo elevato d'errori (failure), si terrà in considerazione la possibilità di rilasciare un numero minore di funzionalità completamente testate. Tali situazioni saranno comunque evitate anche dal fatto che le funzionalità implementate e documentate nel Object Design Document sono già ridotte rispetto al progetto originario.

Sul piano tecnico si cercherà di far sempre un uso intelligente di stub e di driver, in quanto una loro applicazione eccessiva o scorretta potrebbe portare ad appesantimenti o funzionamenti fittizi del sistema.

10.4 Decomposizione gerarchica del sistema

Nell'attività di design è stata effettuata la decomposizione in sottosistemi rimappata sulla base di tre livelli gerarchici:

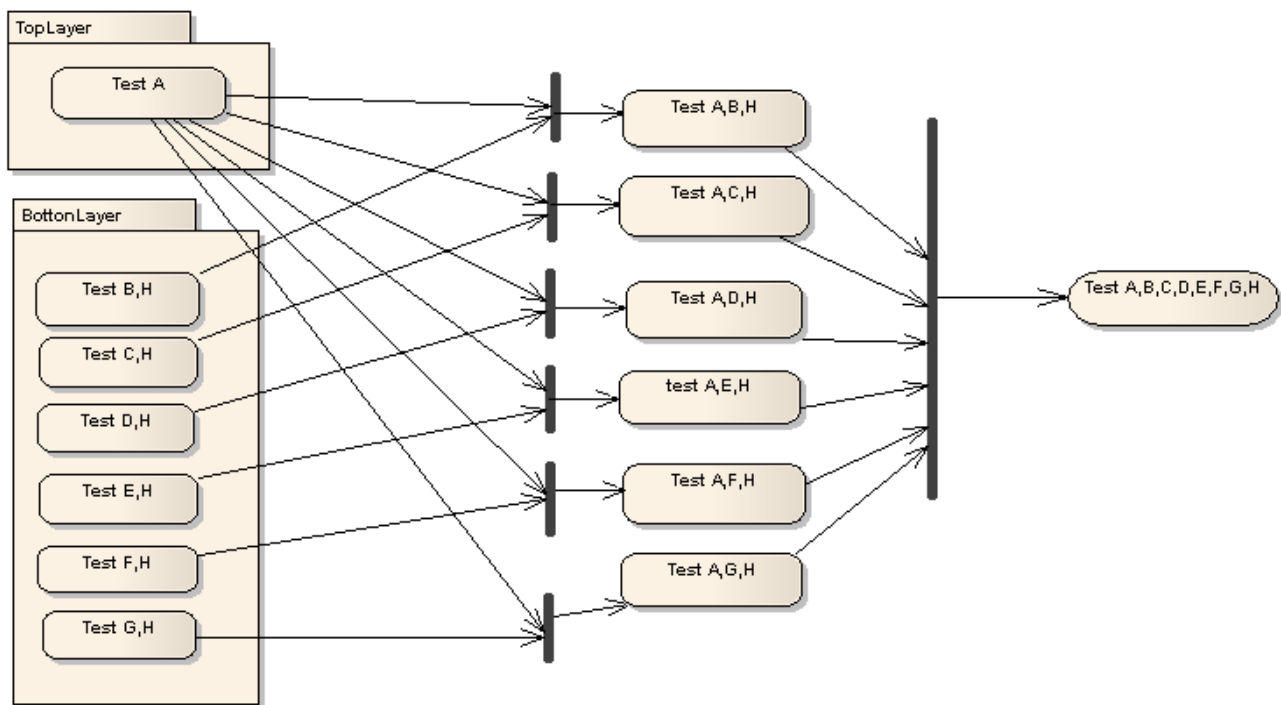
- InterfaceLayer;
- ApplicationLogicLayer;
- StorageLayer.



La comunicazione tra FullDentApplicationLogicLayer e FullDentStorageLayer è trasparente agli occhi dei sviluppatori in quanto è gestita dal JDBC e come si desume dall'ODD i due livelli vengono trattati come uno unico. Tale accorgimento vale anche per il Testing, di conseguenza ciascuna componente verrà testata insieme alla relativa componente di memorizzazione.

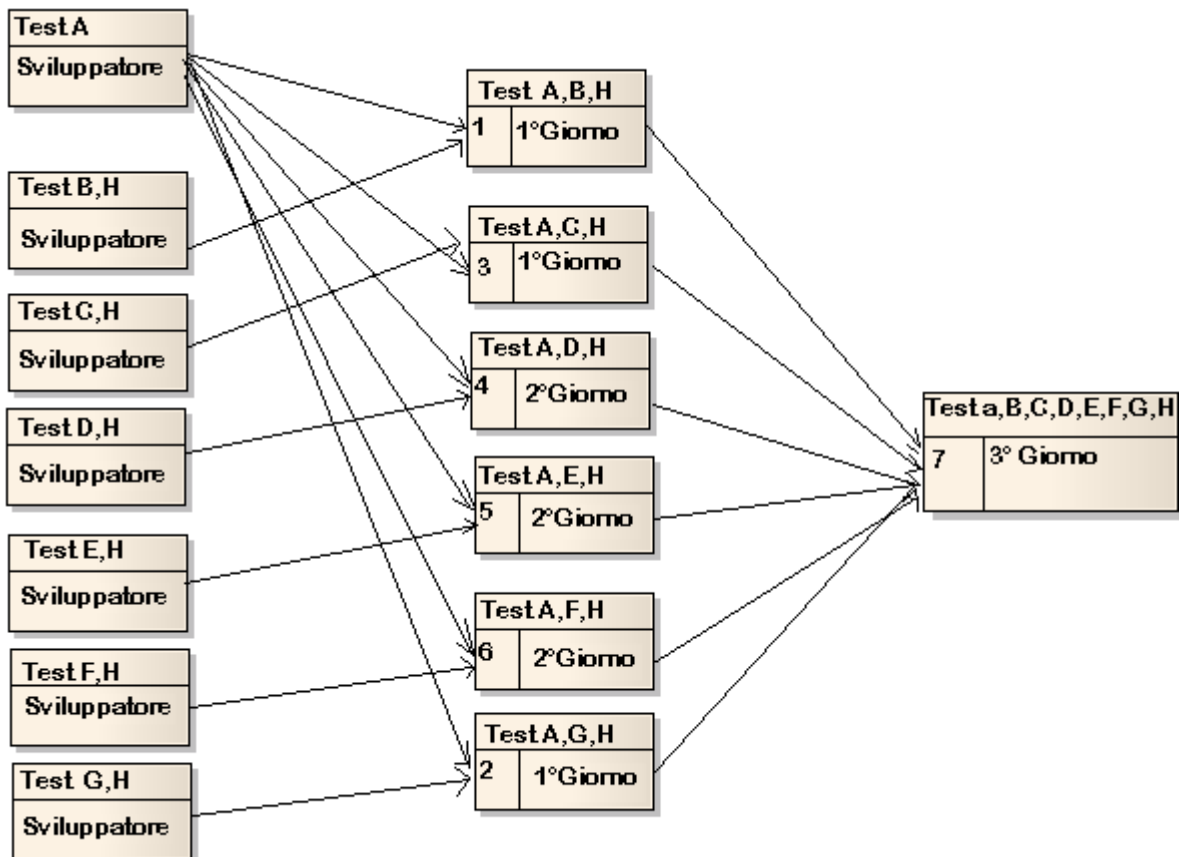
10.5 Organizzazione delle attività di testing

Per far avanzare il test in parallelo, come già detto, sarà utilizzato uno schema a Sandwich per l'organizzazione delle attività. Di seguito ne viene riportata una rappresentazione grafica.



10.6 Schedulazione delle attività di testing

L'attività di Testing, come già accennato, saranno svolte in un massimo di tre giorni. Lo scheduling di tali attività verrà illustrato attraverso il diagramma di PERT, basato sul diagramma delle attività del precedente paragrafo (10.4) e riprodotto nella figura sottostante.



11 Glossario

Termini	Definizione
<i>FullDent</i>	Nome del sistema Software sviluppato
<i>Testing di Integrazione</i>	Fase in cui le singole componenti vengono integrate e testate insieme.
<i>Testing di Sistema</i>	Fase finale del testing, in cui tutti i componenti sono stati integrati e il sistema viene testato nella sua interezza.
<i>Testing d'Unità</i>	Fase in cui ogni componente viene testata singolarmente.
<i>Black Box Testing</i>	Tipo di testing incentrato sul comportamento input/output che trascura la struttura interna delle componenti.
<i>White Box Testing</i>	Tipo di testing incentrato sulla struttura interna della componente. Viene testato ogni stato presente nel modello dinamico dell'oggetto.
<i>Bottom Up Testing</i>	Tipo di testing di integrazione, in cui i componenti vengono progressivamente integrati a partire da quelli di livello più basso.
<i>Top Down Testing</i>	Tipo di testing di integrazione in cui i componenti vengono progressivamente integrati a partire da quelli di livello più alto.
<i>Sandwich Testing</i>	Tipo di testing di integrazione che combina i testing bottom-up e top-down.
<i>Test Stub</i>	Implementazione parziale di una componente da cui la componente testata dipende
<i>Test driver</i>	Implementazione parziale di una componente che dipende da quella testata
<i>Classi d'equivalenza</i>	Insiemi di oggetti con proprietà comuni rispetto ad una funzionalità. Il risultato del testing su un oggetto è valido per l'intera classe di appartenenza.
<i>Layer</i>	Ogni layer (o strato) rappresenta una parte dei sottosistemi risultanti dalla decomposizione effettuata nel System

	Design. Un layer può dipendere solo dai layer al di sotto di lui.
<i>ODD</i>	Acronimo di Object Design Document, documento di progettazione preimplementativa del sistema.
<i>RAD</i>	Acronimo di Requirements Analysis Document, documento di analisi e raccolta dei requisiti
<i>SDD</i>	Acronimo di System Design Document, documento di progettazione del sistema
<i>TP</i>	Acronimo di Test Plan, il presente documento.