



MAINTRANCE ANALYSIS



Riferimento	
Versione	1.6
Data	01/07/2022
Destinatario	Prof.re De Lucia
Presentato da	Vincenzo Esposito, Carmine Laudato



Revision History

Data	Versione	Autori	Descrizione
30/05/2022	1.1	Carmine Laudato	Scopo del documento e panoramica del sistema attuale
08/06/2022	1.2	Vincenzo Esposito	Revisione della documentazione scritta, inserimento dell'analisi riguardante la CR-01 (Starting Impact Set)
09/06/2022	1.3	Vincenzo Esposito, Carmine Laudato	Revisione della documentazione scritta, inserimento dell'analisi riguardante la CR-01 (Candidate Impact Set)
19/06/2022	1.4	Vincenzo Esposito	Inserimento dei componenti del sistema aggiornato inerenti alla CR-01.
29/06/2022	1.5	Vincenzo Esposito	Inserimento dello starting impact set inerenti alla CR-02
30/01/2022	1.6	Vincenzo Esposito, Carmine Laudato	Revisione della documentazione



Sommario

1.	Scopo del documento.....	5
2.	Panoramica del sistema attuale.....	5
2.1	Descrizione del sistema attuale.....	5
2.2	Attori del sistema	6
2.3	Requisiti Funzionali.....	6
RF 1 - Registrazione		6
RF 2 - Login		6
RF 3 - Dashboard.....		6
RF 4 - Genitore.....		6
RF 5 – Segretario.....		7
RF 6 – Responsabile		7
RF 7 – Centro		7
2.4	Architettura del sistema	7
2.5	Database	8
2.5.1	Class Diagram	9
2.6	Testing.....	10
3.	Attività di manutenzione.....	11
3.1	Change Request 01	11
3.1.1	Identificazione, analisi e valutazione.....	12
3.1.2	Impact analysis	13
3.1.2	Sistema aggiornato.....	20
3.2	Change Request 02	23
3.2.2	Identificazione, analisi e valutazione.....	23
3.2.3	Impact analysis	24
3.2.4	Risultati dell’impact analysis	25
3.3	Change Request 03	26
3.3.2	Identificazione, analisi e valutazione.....	26
4.	Sistema modificato	28
4.1	Descrizione del sistema attuale.....	28
4.2	Attori del sistema	29
4.3	Requisiti Funzionali.....	29
RF 1 - Registrazione		29
RF 2 - Login		29
RF 3 - Dashboard.....		29



Laurea Magistrale in informatica -
Università di Salerno Corso di
*Ingegneria, Gestione ed Evoluzione del
Software* Prof. De Lucia Andrea



RF 4 - Genitore	29
RF 5 – Segretario.....	30
RF 6 – Responsabile	30
RF 7 – Centro	30
4.4 Architettura del sistema	30
4.4.1 Database.....	31
4.4.2 Class Diagram	32



1. Scopo del documento

In questo documento saranno descritte tutte le nuove funzionalità implementate, in riferimento al documento "Change Requests".

Si andrà ad analizzare come tale modifica impatterà sugli artefatti del sistema esistente, osservando il rapporto costi/benefici e i possibili rischi derivati dalla fase di progettazione.

Questo documento include anche uno studio di fattibilità.

2. Panoramica del sistema attuale

2.1 Descrizione del sistema attuale

Il sistema è una web application "Summer Center" volta all'automatizzazione della compilazione dei moduli riguardanti i genitori, i bambini, l'iscrizione e alla gestione della segreteria per risolvere i problemi dell'errata compilazione dei moduli e del sistema di pagamento.

L'applicazione offre al genitore le seguenti funzionalità:

- accedere ad un'apposita dashboard dove può visualizzare tutti i dettagli dei propri dati anagrafici, dei bambini e dell'iscrizione;
- compilare gli appositi moduli anagrafici e informativi;
- iscrizione dei bambini al centro con la possibilità di finalizzare le iscrizioni con un sistema di pagamento online affidabile.
- visualizzare i dettagli delle varie iscrizioni effettuate;
- modificare i propri dati personali;
- decidere di pagare un'iscrizione istantaneamente o successivamente.

Le funzionalità offerte alla segreteria sono:

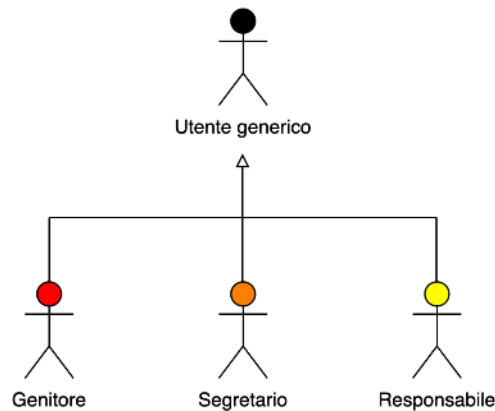
- visualizzare tutte le informazioni delle iscrizioni effettuate dai genitori;
- scegliere la percentuale di rimborso da attribuire ad un'iscrizione disdetta;
- controllare la validità dell'iscrizione mostrando al sistema il QR Code associato all'iscrizione.

Le funzionalità che il sistema offre al responsabile sono:

- registrare diversi segretari al sistema;
- visualizzare l'andamento delle iscrizioni del centro attraverso grafici e calcoli statistici;



2.2 Attori del sistema



2.3 Requisiti Funzionali

RF 1 - Registrazione

- **RF 1.0:** Quando il genitore vuole accedere per la prima volta al sistema, Summer Center deve permettere la registrazione dell'account e la verifica dell'account, per poter effettuare successivamente l'accesso al sistema stesso.
- **RF 1.1:** Quando il responsabile vuole attivare un segretario, il sistema deve permettere la registrazione del segretario da parte del responsabile.

RF 2 - Login

- **RF 2.0:** Quando il responsabile o il genitore o il segretario vogliono accedere alle proprie dashboard, il sistema deve permettere l'accesso tramite e-mail e password.

RF 3 - Dashboard

- **RF 3.0:** Il sistema deve permettere la visualizzazione di dashboard e funzionalità diverse in base all'utente loggato.

RF 4 - Genitore

- **RF 4.0:** Il genitore può effettuare un'iscrizione solo dopo aver compilato i propri dati personali obbligatori.
- **RF 4.1:** Il genitore può effettuare il pagamento dell'iscrizione tramite il metodo di pagamento con Carta di credito/Carta di debito o PayPal.
- **RF 4.2:** Il genitore può effettuare la prenotazione selezionando una o più settimane (la settimana va dal lunedì al venerdì) dei mesi che vanno da Maggio a Settembre.
- **RF 4.3:** Il genitore può modificare i propri dati personali.
- **RF 4.4:** Il genitore può visualizzare tutte le iscrizioni da lui effettuate (pagate e non).



- **RF 4.5:** Il genitore può visualizzare i dettagli di una singola iscrizione.
- **RF 4.6:** Il genitore può disdire un'iscrizione.
- **RF 4.7:** Il genitore può cambiare la password per i suoi prossimi accessi.
- **RF 4.8:** Il genitore può decidere di pagare un'iscrizione istantaneamente o successivamente. Decidendo di pagarla successivamente potrà accedere ad una pagina in cui sono presenti le iscrizioni che può selezionare e pagare.

RF 5 – Segretario

- **RF 5.0:** Il segretario può visualizzare tutte le iscrizioni effettuate dai genitori.
- **RF 5.1:** Il segretario può visualizzare i dettagli di una singola iscrizione.
- **RF 5.2:** Il segretario può scegliere la percentuale di rimborso da attribuire ad un'iscrizione disdetta.
- **RF 5.3:** Il segretario può controllare la validità dell'iscrizione mostrando al sistema il QR Code associato all'iscrizione.
- **RF 5.4:** Il segretario deve obbligatoriamente cambiare password al primo accesso al sistema.
- **RF 5.5:** Il segretario può cambiare la password per i suoi prossimi accessi.

RF 6 – Responsabile

- **RF 6.0:** Il responsabile può registrare segretari al sistema.
- **RF 6.1:** Il responsabile può visualizzare l'andamento delle iscrizioni del centro attraverso grafici e calcoli statistici.
- **RF 6.2:** Il responsabile può cambiare la password per i suoi prossimi accessi.

RF 7 – Centro

- **RF 7.0:** Il centro può sostenere massimo 70 bambini nello stesso periodo.

2.4 Architettura del sistema

Il sistema attuale, per garantire i servizi, utilizza un'architettura di tipo Client-Server. Il Client è rappresentato dal Browser utilizzato dall'utente.

Come Web Server è utilizzato Apache Tomcat 9.0, dove le pagine JSP/HTML, JS e CSS realizzeranno la UI del sistema e le Java Servlet la logica applicativa. Il client comunica con il Web Server attraverso il protocollo di comunicazione HTTPS.

Il server può comunicare con il database, attraverso le API di JDBC. La gestione dei dati persistenti avviene attraverso il DBMS MySQL.

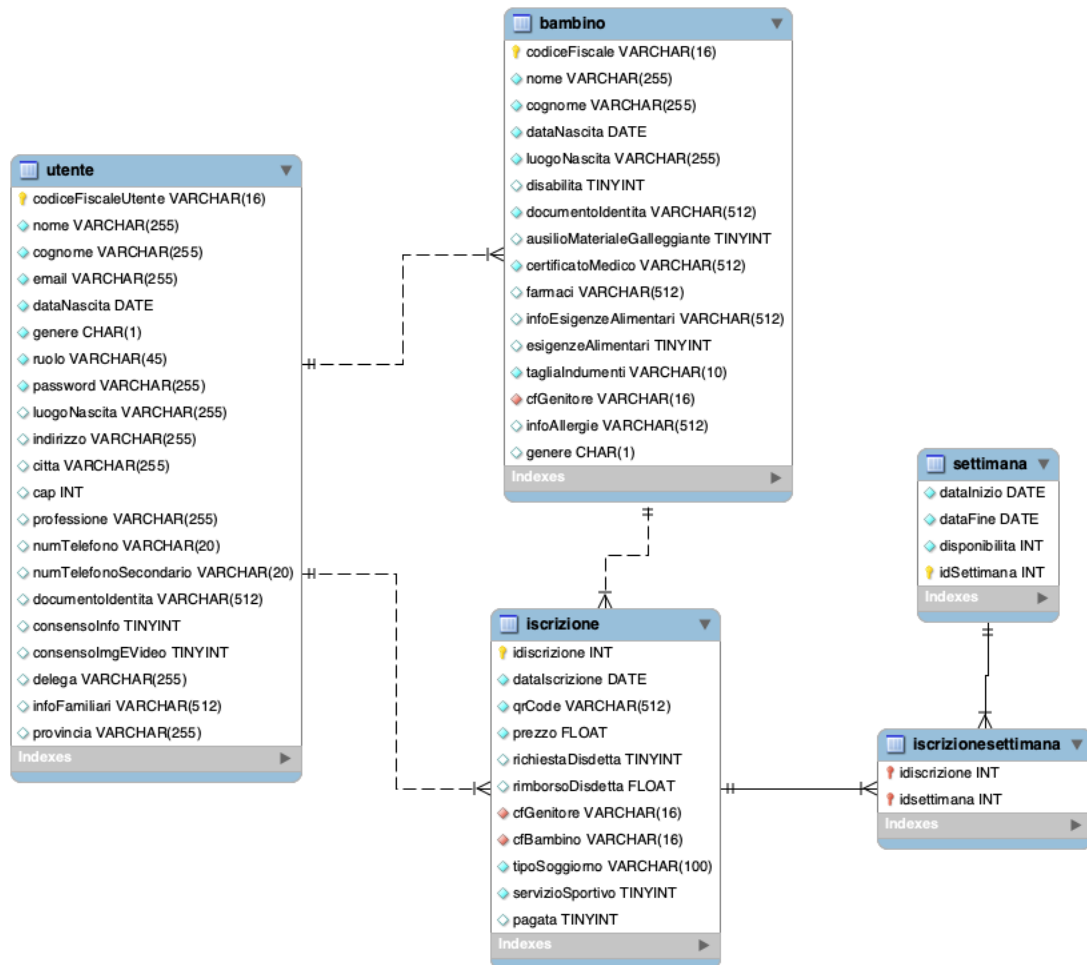
Attraverso il diagramma delle classi abbiamo individuato le entità che saranno memorizzate nel database. Nel nostro caso tutte le entità saranno rese persistenti in un database.

Il controllo globale del software è di tipo Event-Driven Control. Il ciclo principale attende un evento esterno, quando l'evento si verifica viene spedito all'oggetto appropriato. Il controllo risiede in un dispatcher che invoca i servizi di un sottosistema.



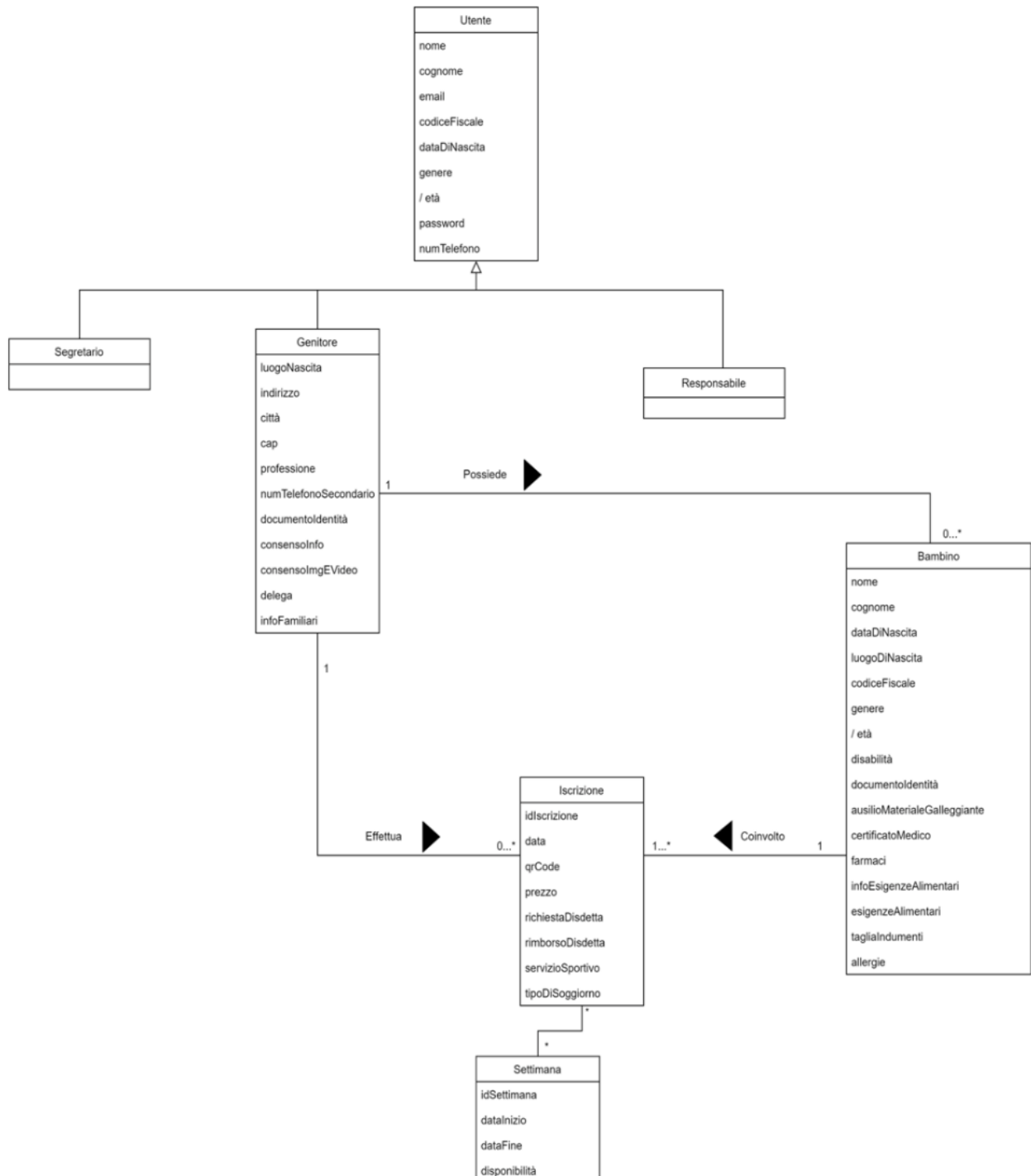
Il sistema è implementato, lato client-side mediante il pattern MVC ereditato dal framework utilizzato, lato server-side si necessita di un qualsiasi browser capace di interpretare il linguaggio JavaScript (al fine di permettere l'interazione con l'interfaccia del sistema).

2.5 Database





2.5.1 Class Diagram





2.6 Testing

Il sistema è stato sottoposto a:

- Testing di unità (approccio black-box) in cui è stata testata ogni singola funzione degli oggetti creati.
- Testing di sistema per dimostrare che i requisiti commissionati dal cliente sono soddisfatti.

Il testing di integrazione (approccio bottom-up), non è stato effettuato per problemi legati ad interferenze tra i vari componenti, le quali non hanno concesso la connessione con il DB.

Sono state testate le funzionalità usate per l'autenticazione di un utente al sistema.



3. Attività di manutenzione

Durante l'attività di manutenzione sono state considerate le change requests ottenute dai feedback sia degli utenti sia degli sviluppatori.

Lo Starting Impact Set rappresenta una prima analisi delle varie classi che devono essere modificate per il completamento delle change request.

Di seguito verranno riportate la suddivisione che abbiamo effettuato per rappresentare l'impatto che la modifica avrà su quelle determinate classi:

- FORTE: nel caso in cui sia necessario riscrivere completamente l'artefatto.
- MEDIO: nel caso in cui sia necessario una rivisitazione e modifica di un artefatto.
- DEBOLE: nel caso in cui siano necessarie solo modifiche marginali.
- NULLO: nel caso che non sono presenti modifiche

3.1 Change Request 01

Change Request	Data	Richiedente	Priorità
1	29/04/2022	Sviluppatori	Alta
Descrizione			
<p>Per il sistema attuale si vuole aggiungere un nuovo requisito che consiste nel dare la possibilità agli utenti (genitori) di iscrivere i propri bambini selezionando un centro estivo dalla lista di centri estivi dislocati per il territorio e presenti sul sistema.</p> <p>Questa CR deriva da feedback ottenuti dagli utenti (genitori) e da noi sviluppatori testando la web application.</p>			
Scopo			
<p>Attualmente la web application Summer Center consente di effettuare prenotazioni ai bambini da parte dei genitori ad un unico centro estivo.</p> <p>L'obiettivo della CR è quello di espandere la piattaforma ad un dominio pluri-centro per consentire la prenotazione dei bambini a più centri estivi dislocati per il territorio.</p> <p>L'utente Genitore, potrà:</p> <ul style="list-style-type: none">• Iscrivere un bambino/i alla piattaforma;• Selezionare il campo estivo dove si vuole effettuare la prenotazione. <p>Le modifiche riguarderanno la sostituzione del requisito "RF 7.0". Il nuovo requisito comprenderà l'inserimento delle varie informazioni per l'aggiunta di un nuovo centro estivo alla piattaforma.</p>			
Conseguenza se non accettata			
Il sistema non cambierebbe, ma il dominio di utilizzo risulterebbe ristretto.			



3.1.1 Identificazione, analisi e valutazione

3.1.1.1 Identificazione, descrizione e valutazione dei costi

Identificatore	Valutazione	Motivazione
Integrazione dei requisiti funzionali	Forte	L'aggiunta del requisito comporta qualche modifica anche in altre componenti.
Modifica DB	Forte	Dovranno essere aggiunte due o più tabelle per realizzare la seguente Change Request.
Implementazione con Java e Javascript	Forte	Lo sviluppatore ha sufficienti conoscenze della tecnologia, ma impiegherà diverso tempo per applicare le modifiche poiché riguarda l'intero funzionamento del sistema.
Testing Funzionale	Forte	Il testing richiede una modifica di tutti i test cases relativi all'iscrizione di un bambino, alla visualizzazione di un'iscrizione e la creazione di nuovi test cases per le nuove funzionalità che verranno aggiunte per il responsabile (operazioni di inserimento/modifica/eliminazione di un centro estivo).

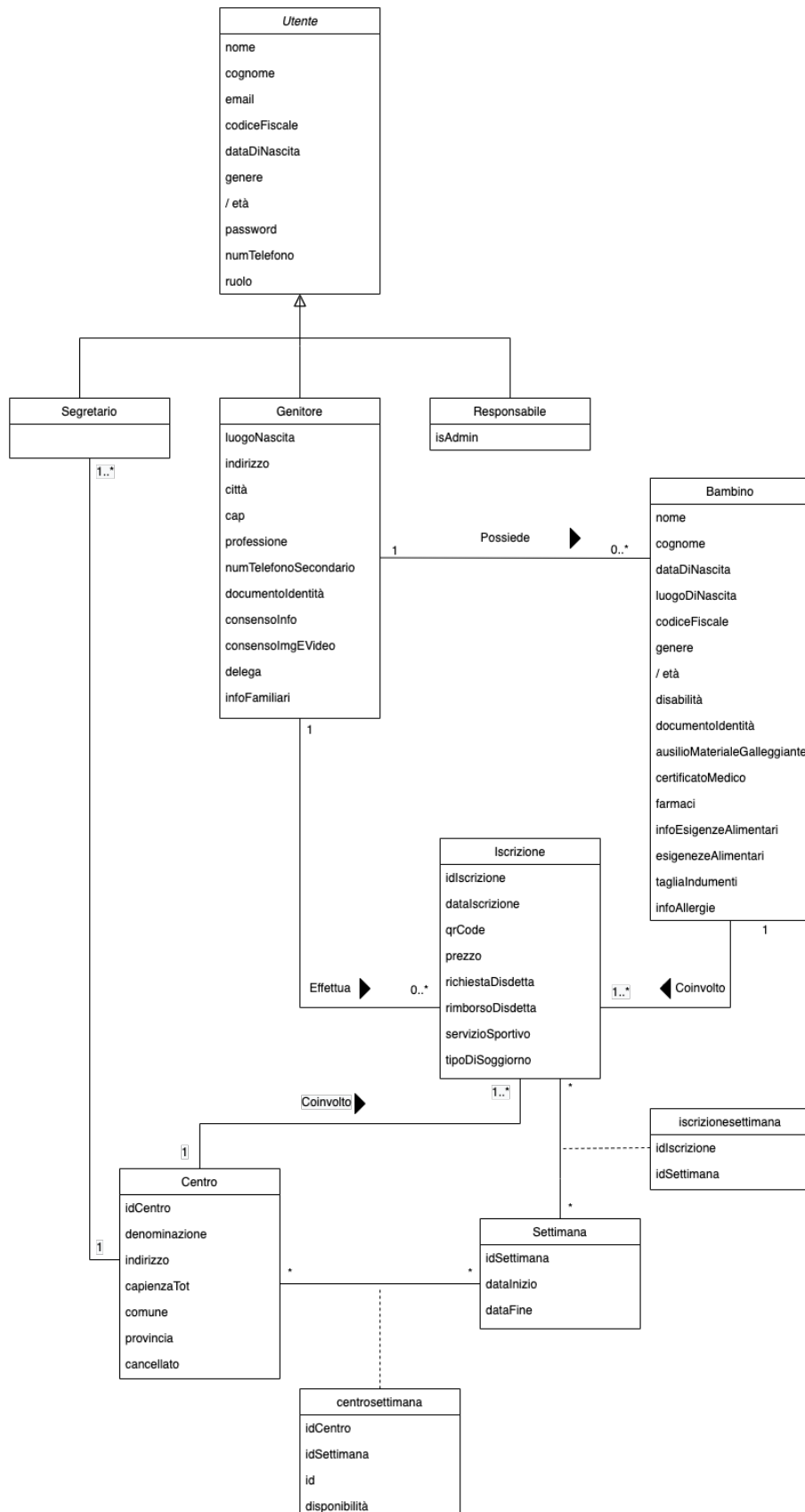
3.1.1.2 Identificazione, descrizione e valutazione dei benefici

Identificatore	Valutazione	Motivazione
Possibilità di iscrivere un bambino ad un altro centro estivo	Forte	Il genitore può finalmente iscrivere il proprio bambino, attraverso la piattaforma summer center a diversi centri estivi.

3.1.1.3 Identificazione, descrizione e valutazione dei rischi

Identificatore	Valutazione	Motivazione
Introduzione di nuovi fault	Medio	L'attività di testing da effettuare potrà avere una media probabilità di riscontrare fault, questo è dovuto al fatto che il sistema è già stato sottoposto ad una fase di testing.
Carico di lavoro eccessivo per il client	Debole	Il client dovrà solo elaborare dei dati provenienti dai diversi centri estivi.
Carico di lavoro eccessivo per il server	Debole	Il server dovrà garantire la persistenza dei dati relativi ai diversi centri e alle informazioni legate alle varie iscrizioni.

3.1.2 Impact analysis





3.1.2.1 Starting Impact Set

Per quanto concerne lo Starting Impact Set, siamo partiti con l'analizzare gli artefatti presenti all'interno del Requirements Analysis Document (RAD) nello specifico utilizzando il diagramma delle classi per capire l'impatto che avessero i nuovi componenti da aggiungere.

Come possiamo notare dalla Figura alla pagina precedente, sono state inserite delle nuove entità al suo interno consentendoci di andare a capire come il nostro sistema sarebbe mutato. Le modifiche impatteranno sulla maggior parte del nostro sistema poiché la gestione dell'iscrizione è il fulcro di esso.

Innanzitutto, abbiamo valutato i requisiti funzionali presenti e abbiamo appurato che l'impatto della nostra modifica sarà forte. Le modifiche riguarderanno:

- Modifica del requisito funzionale 4.2, il genitore può effettuare la prenotazione selezionando una o più settimane (la settimana va dal lunedì al venerdì) nei mesi compresi tra Maggio e Settembre e può selezionare il centro estivo in cui vuole iscrivere il proprio bambino. Tale modifica è stata evinta dall'impatto che ha avuto l'inserimento dell'entità centro all'interno del sistema. L'entità sarà coinvolta direttamente nelle iscrizioni poiché un'iscrizione sarà legata ad uno ed un solo centro e un centro sarà coinvolto in uno o più iscrizioni.
- Modifica del requisito funzionale 5.0, il segretario può visualizzare tutte le iscrizioni effettuate dai genitori per il centro in cui esso è coinvolto. Tale modifica è stata evinta dall'impatto che ha avuto l'inserimento dell'entità centro all'interno del sistema. L'entità sarà coinvolta direttamente con il segretario in quanto un segretario può gestire le operazioni di uno ed un solo centro ma un centro può essere gestito da uno o più segretari.
- Modifica del requisito funzionale 6.0, Il responsabile può registrare segretari al sistema associandoli ad un determinato centro estivo. Tale modifica è stata evinta dall'impatto che ha avuto l'inserimento dell'entità centro all'interno del sistema.
- Modifica del requisito funzionale 7.0, il sistema da mono-centro diventerà pluri-centro e ogni centro potrà ospitare al più 100 bambini a settimana. La modifica è una conseguenza diretta dell'aggiunta dell'entità centro.
- Inserimento del requisito funzionale che consenta di aggiungere un nuovo centro estivo alla piattaforma da parte del responsabile, esso è scaturito dall'impatto che ha avuto l'inserimento dell'entità centro all'interno del sistema.
- Inserimento di un requisito che consenta la di effettuare le operazioni di cancellazione di un centro estivo dalla piattaforma, esso è scaturito dall'impatto che ha avuto l'inserimento dell'entità centro all'interno del sistema.

Dopodiché, abbiamo valutato i casi d'uso suddividendoli per i diversi utenti che fanno parte del nostro sistema. Abbiamo appurato che l'impatto della nostra modifica sarà forte.

Le modifiche riguarderanno:

1. Analisi (Genitore):

- Modifica del caso d'uso UC_Iscrizione_Bambino, dovrà essere inserita la casella per effettuare la scelta del centro estivo in cui si vuole iscrivere il proprio bambino.
- Modifica del caso d'uso UC_VisualizzaIscrizioni, dovrà essere inserita la voce che indica il centro estivo selezionato per quella determinata iscrizione.



- Modifica del caso d'uso UC_VisualizzaDettagliIscrizione, dovrà essere inserita la voce che indica il centro estivo selezionato per quella determinata iscrizione.
2. Analisi (Segretario):
- Modifica del caso d'uso UC_ApprovaDisdetta, dovrà essere inserita la voce che indica il centro estivo selezionato per quella determinata iscrizione che il segretario deve disdire. Generalmente un segretario può disdire solo le iscrizioni che fanno parte del centro estivo di appartenenza.
 - Modifica del caso d'uso UC_VisualizzaIscrizionePagate, dovrà essere inserita la voce che indica il centro estivo selezionato per le iscrizioni pagate dai Genitori.
 - Modifica del caso d'uso UC_RicercaIscrizioni, dovrà essere inserita la voce che consentirà di ricercare le iscrizioni in base al centro estivo di appartenenza.
 - Modifica del caso d'uso UC_VisualizzaDettagliIscrizionePagata, dovrà essere inserita la voce che indica il centro estivo selezionato per quella determinata iscrizione che il segretario decide di visualizzare.
3. Analisi (Responsabile):
- Modifica del caso d'uso UC_AndamentoCentro, dovrà essere inserito un grafico all'interno della dashboard del responsabile che rappresenta il numero di iscrizioni per i centri presenti sulla piattaforma.
 - Modifica del caso d'uso UC_RegistraSegretario, il quale consentirà al responsabile di inserire un nuovo segretario al sistema associandolo ad un determinato centro.
 - Inserimento di un nuovo caso d'uso UC_InserisciCentro, il quale consentirà al responsabile di inserire un nuovo centro estivo all'interno del sistema.
 - Inserimento di un nuovo UC_EliminaCentro, il quale consentirà al responsabile di eliminare logicamente dal sistema un centro estivo.

A seguire, abbiamo valutato il database esistente, appurando anche in questo caso che le modifiche avranno un impatto forte. Le modifiche riguarderanno:

- Un'entità denominata "Centro", la quale conterrà tutte le informazioni dei centri estivi dislocati sul territorio;
- Un nuovo attributo denominato "id_centro" nelle entità relativa all'iscrizione, la quale indicherà la relazione che vi è tra l'entità centro e l'entità iscrizione, ovvero un centro può permettere una o più iscrizioni ma un centro può essere legata ad una e sola iscrizione.
- Una relazione molti a molti tra centro e settimana, la quale indicherà la relazione che vi è tra l'entità centro e settimana, ovvero un centro può contenere una o più settimane e una settimana può essere correlata ad uno o più centri. La nuova entità sarà denominata "centrosettimana".

Infine, abbiamo analizzato le classi esistenti, considerando un impatto forte della modifica. Infatti, le modifiche riguarderanno la maggior parte del sistema esistente:

- Modifica nel package entity/Iscrizione.java andando ad inserire il nuovo campo id_centro per identificare il centro associato ad essa e una nuova query che ci consenta di ricercare le iscrizioni relative ad un centro estivo.
- Modifica nel package dao/IscrizioneManage.java andando ad inserire all'interno dell'interfaccia la nuova query per ricercare le iscrizioni relative ad un centro estivo.



- Modifica nel package dao/IscrizioneManageDS.java andando ad inserire all'interno dell'implementazione dell'interfaccia la nuova query per ricercare le iscrizioni relative ad un centro estivo.
- Modifica nel package control.GestioneIscrizione della classe IscrizioneControl.java andando ad inserire:
 - il collegamento logico tra le varie settimane e i vari centri per la gestione della disponibilità;
 - una lista contenente i centri estivi presenti all'interno del sistema;
 - gestione della scelta del centro per la creazione di una nuova iscrizione.
- Modifica nel package WebContent/iscrizione.jsp andando ad inserire la scelta del centro con le relative settimane disponibili.
- Modifica nel package WebContent/Dettagli_iscrizione.jsp andando ad inserire nella parte grafica la visualizzazione del centro estivo in cui è stato iscritto il bambino.
- Modifica nel package WebContent/List_iscrizioni.jsp andando ad aggiungere la visualizzazione delle iscrizioni relative ad un determinato centro estivo se viene selezionato il centro estivo in Gestione Centri.
- Modifica nel package control/VisualizzaIscrizioniControl.java andando a raggruppare in diversi oggetti le iscrizioni in base ai centri.
- Modifica nel package entity/Settimana.java andando a rimuovere l'attributo disponibilità.
- Modifica nel package dao/SettimanaManage.java andando a sostituire la dichiarazione nell'interfaccia relativo alle settimane disponibili con una dichiarazione che restituisce tutte le settimane.
- Modifica nel package dao/SettimanaManageDS.java andando a sostituire la query delle settimane disponibili con una query che restituisce tutte le settimane.
- Modifica della classe VisualizzaIscrizioniControl.java del package gestioneIscrizione andando a inserire la lista delle iscrizioni in base al centro selezionato in Gestione Centri.
- Modifica nel package WebContent/header.jsp andando ad inserire un controllo per inserire le funzionalità da parte del responsabile del sistema, il quale potrà inserire o eliminare i centri estivi dal sistema.

3.1.2.2 Candidate Impact Set

In seguito alla fase di analisi eseguita nello Starting Impact Set, abbiamo preso in considerazione la correlazione presente tra i casi d'uso fino ad arrivare alle classi sviluppate. Questo lavoro è stato fatto per individuare le componenti impattate indirettamente da quelle prese in considerazione nel SIS (Starting Impact Set) descritto sopra.

Partendo dal nostro SIS, abbiamo analizzato più approfonditamente il sistema, evidenziando le varie dipendenze per poi farne un'analisi ed estrapolare le componenti che faranno parte del nostro CIS (Candidate Impact Set).

Il nostro Candidate Impact Set è quasi del tutto vuoto, perché la modifica ha impattato la funzione principale del sistema e di conseguenza tutte le componenti del sistema che sono state modificate erano legate strettamente all'iscrizione che per il nostro sistema è la funzionalità principale.



Classe	Impatto	Descrizione
iscrizione.js	Diretto	La modifica concerne l'inserimento di controlli in javascript per caricare in maniera dinamica le settimane disponibili per un determinato centro selezionato all'interno del form.

3.1.2.3 Actual Impact Set

Dopo aver completato il Candidate Impact Set, abbiamo implementato le varie modifiche che erano state descritte nelle due fasi precedenti. Le modifiche sono state suddivise in modifiche effettive e modifiche false positive. Con quest'ultime indichiamo tutte le modifiche che sono state descritte nelle fasi precedenti ma che non sono servite per l'implementazione della change request.

3.1.2.3.1 Modifiche effettive

Classe	Impatto	Descrizione
Iscrizione.java	Diretto	Come descritto nella prima fase dell'impact analysis la modifica di questa classe concerne nell'aggiunta di una nuova variabile che indica il nuovo campo id_centro presente nell'entità relativa alle iscrizioni. Inoltre, è stata aggiunta una nuova query che ci consente di ricercare le iscrizioni relative ad un centro estivo.
IscrizioneManage.java	Diretto	La modifica di questa interfaccia IscrizioneManage.java è stata eseguita inserendo al suo interno la dichiarazione del metodo che rappresenta la query per effettuare la ricerca delle iscrizioni relative ad un centro estivo. Il metodo restituisce la lista delle iscrizioni e prende come parametro il codice identificativo del centro estivo.
IscrizioneManageDS.java	Diretto	La modifica della classe che implementa l'interfaccia descritta precedentemente consiste nell'implementazione del metodo dichiarato nell'interfaccia. Esso consiste nella ricerca e nella restituzione delle iscrizioni relative ad un centro estivo.
IscrizioneControl.java	Diretto	Questa classe ha subito le seguenti modifiche:



		<ul style="list-style-type: none">• restituzione alla pagina jsp della lista contenente i centri estivi presenti all'interno del sistema;• l'inserimento del collegamento logico tra le varie settimane e i vari centri per la gestione della disponibilità restituendo alla pagina jsp dell'iscrizione le informazioni aggiornate;• modifica delle varie componenti per la creazione di una nuova iscrizione aggiungendo il campo relativo al centro selezionato.
Iscrizione.jsp	Diretto	La pagina jsp relativa alla creazione di una nuova iscrizione è stata modificata con l'inserimento della scelta del centro tra quelli inseriti nel sistema e non disabilitati e delle relative settimane disponibili.
Iscrizione.js	Diretto	La modifica concerne l'inserimento di controlli in javascript per caricare in maniera dinamica le settimane disponibili per un determinato centro selezionato all'interno del form.
Dettagli_iscrizione.jsp	Diretto	Questa jsp è stata mantenuta andando a visualizzare il centro estivo in cui è stato iscritto il bambino.
List_iscrizioni.jsp	Diretto	Questa classe è stata mantenuta inserendo la possibilità di visualizzare le iscrizioni relative ad un determinato centro estivo.
Settimana.java	Diretto	La modifica riguarda la rimozione dell'attributo legato alla disponibilità sia dalla entity Settimana sia dalla classe che rappresenta l'entity. Essa è stata necessaria in quanto non vi era la possibilità di gestire la disponibilità per le diverse settimane di ogni centro.
SettimanaManage.java	Diretto	Questa interfaccia è stata mantenuta con la sostituzione della dichiarazione relativa alle settimane disponibili con una che restituisce tutte le settimane presenti nel database.
SettimanaManageDS.java	Diretto	Le modifiche effettuate riguardano la sostituzione della query delle settimane



		disponibili con una query che restituisce tutte le settimane.
Header.jsp	Diretto	Le modifiche effettuate riguardano l'aggiunta della visualizzazione delle funzionalità implementate per l'utente Responsabile relative alla gestione dei Centri.
VisualizzaIscrizioniControl.java	Diretto	Le modifiche effettuate riguardano l'aggiunta della visualizzazione delle iscrizioni per un determinato centro.

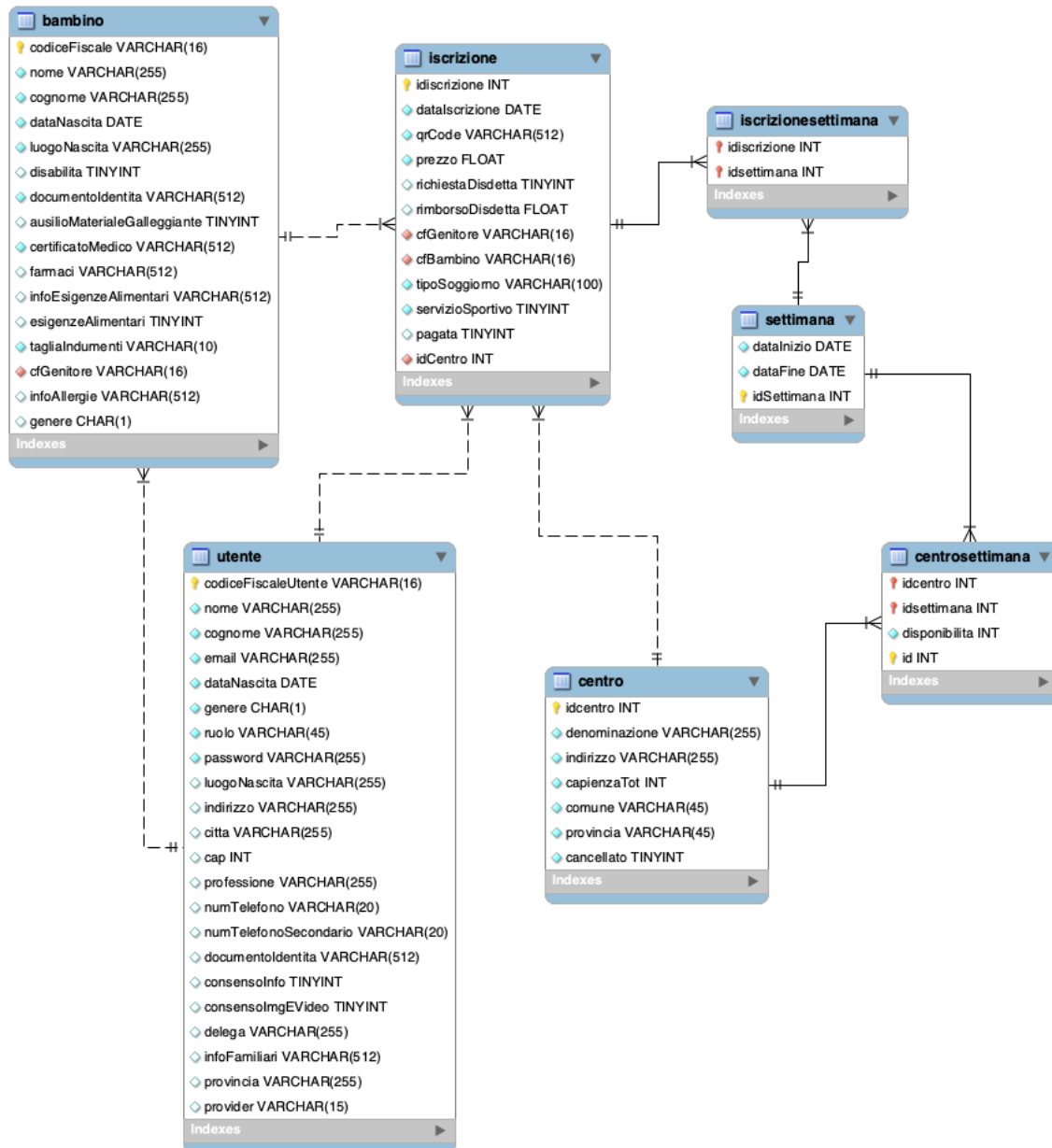
3.1.2.3.2 Fake Positive Impact Set

Classe	Impatto	Descrizione
SingolaIscrizioneControl.java	Nessun impatto	La modifica in questo caso non è stata apportata in quanto in SingolaIscrizioneControl.java viene utilizzato direttamente l'oggetto iscrizione nel quale vi è presente l'id_centro, da esso possiamo risalire a quale centro estivo è stata effettuata l'iscrizione.



3.1.2 Sistema aggiornato

3.1.2.1 Database





3.1.2.2 Componenti aggiunte

Classe	Descrizione
Centro.java	L'aggiunta della seguente classe nel package entity/Centro.java ci ha permesso di rendere la web application pluri-centro in quanto con essa un utente Genitore può scegliere a suo piacimento in quale centro estivo presente nel sistema iscrivere il proprio bambino. La seguente entity contiene le informazioni inerenti al centro estivo e un campo denominato "cancellato" il quale consente un'eliminazione logica dal sistema da parte dell'utente responsabile.
CentroManage.java	L'aggiunta della seguente interfaccia nel package model.dao/CentroManage.java ci ha permesso di definire i metodi che ci consentono di realizzare le seguenti query: <ul style="list-style-type: none">• ricercare un centro mediante il suo id o denominazione;• restituire la lista di tutti i centri;• inserire un nuovo centro estivo;• settare la cancellazione del centro estivo.
CentroManageDS.java	L'aggiunta della seguente interfaccia nel package model.dao/CentroManageDS.java ci ha permesso di implementare i metodi che ci consentono di realizzare le seguenti query: <ul style="list-style-type: none">• ricercare un centro mediante il suo id o denominazione;• restituire la lista di tutti i centri;• inserire un nuovo centro estivo;• settare la cancellazione del centro estivo.
CentroSettimana.java	La seguente classe è stata aggiunta nel package entity/CentroSettimana.java e contiene le informazioni inerenti l'idCentro, l'idSettimana e la disponibilità effettiva per la settimana correlata al centro estivo.



CentroSettimanaManage.java	<p>L'aggiunta della seguente interfaccia nel package <code>model.dao/CentroSettimanaManage.java</code> ci ha permesso di definire i metodi che ci consentono di realizzare le seguenti query:</p> <ul style="list-style-type: none">• ricercare una settimana mediante il suo id;• restituire la lista di tutte le settimane disponibili;
CentroSettimanaManageDS.java	<p>L'aggiunta della seguente interfaccia nel package <code>model.dao/CentroSettimanaManage.java</code> ci ha permesso di implementare i metodi che ci consentono di realizzare le seguenti query:</p> <ul style="list-style-type: none">• ricercare una settimana mediante il suo id;• restituire la lista di tutte le settimane disponibili;
Responsabile.java	<p>Questa classe è stata aggiunta per caratterizzare l'utente di tipo Responsabile, che ha permessi di Admin del sistema.</p>
CentriControl.java	<p>La seguente servlet è stata aggiunta nel package <code>control.gestioneCentri/CentriControl.java</code> e ci consente di gestire l'eliminazione di un centro estivo da parte di un utente Responsabile.</p>
VisualizzaCentriControl.java	<p>La seguente servlet è stata aggiunta nel package <code>control.gestioneCentri/VisualizzaCentriControl.java</code> e ci consente di visualizzare i centri estivi presenti sulla piattaforma inseriti da parte di un utente Responsabile.</p>
list_centri.jsp	<p>La pagina jsp relativa alla visualizzazione, gestione ed eliminazione di un centro estivo da parte dell'utente Responsabile.</p>
centri.jsp	<p>La pagina jsp relativa all'inserimento di un centro estivo da parte dell'utente Responsabile al sistema.</p>



3.2 Change Request 02

Change Request	Data	Richiedente	Priorità
2	29/04/2022	Feedback (Genitori)	Media
Descrizione			
Dopo una prima analisi dei feedback ricevuti dai vari genitori si è deciso di voler modificare il requisito funzionale 2.0 che riguarda il login di un utente al sistema. Attualmente un genitore può loggarsi al sistema mediante la sua e-mail e password ma non mediante l'utilizzo di servizi esterni come Google o Facebook. Questa CR deriva dai feedback raccolti da utenti (genitori) che hanno utilizzato la web application.			
Scopo			
L'obiettivo di questa CR è quello di permettere a chi utilizza la web application Summer Center di scegliere al momento del login se effettuare l'accesso mediante il suo indirizzo e-mail oppure mediante i servizi di autenticazione di Google o Facebook. Le modifiche riguarderanno l'ampliamento del RF 2.0 il quale riguarda l'accesso al sistema.			
Conseguenza se non accettata			
Il sistema non cambierebbe, ma non soddisferebbe pienamente i feedback dei vari utenti.			

3.2.2 Identificazione, analisi e valutazione

3.2.2.1 Identificazione, descrizione e valutazione dei costi

Identificatore	Valutazione	Motivazione
Integrazione dei requisiti funzionali	Medio	La modifica del requisito 2.0 comporta qualche modifica nelle altre componenti.
Modifica DB	Debole	Aggiunta di qualche campo per salvarci le informazioni del provider dell'OAuth2.0
Implementazione con Java e Javascript	Forte	Lo sviluppatore ha sufficienti conoscenze della tecnologia, ma dovrà impiegare diverso tempo per integrare i componenti che consentono l'utilizzo dell'OAuth2.0.
Testing Funzionale	Forte	Il testing richiede una modifica di tutti i test cases già esistenti relativi al login dell'utente alla piattaforma Summer Center.



3.2.2.2 Identificazione, descrizione e valutazione dei benefici

Identificatore	Valutazione	Motivazione
Possibilità di velocizzare l'operazione di login	Forte	Il genitore può velocizzare l'operazione di login attraverso l'utilizzo di sistemi esterni grazie all'utilizzo dell'OAuth 2.0 e quindi non deve necessariamente ricordarsi username e password.

3.2.2.3 Identificazione, descrizione e valutazione dei rischi

Identificatore	Valutazione	Motivazione
Introduzione di nuovi fault	Debole	L'attività di testing da effettuare potrà avere una bassa probabilità di riscontrare fault, questo è dovuto al fatto che il sistema è già stato sottoposto ad una fase di testing.
Carico di lavoro eccessivo per il client	Debole	Il client dovrà solo elaborare dei dati provenienti dai diversi centri estivi.
Carico di lavoro eccessivo per il server	Debole	Il server dovrà garantire la persistenza dei dati relativi ai diversi centri e alle informazioni legate alle varie iscrizioni.

3.2.3 Impact analysis

3.2.3.1 Starting Impact Set

Per quanto concerne lo Starting Impact Set, siamo partiti con l'analizzare gli artefatti presenti all'interno del Requirements Analysis Document (RAD) nello specifico utilizzando il diagramma delle classi per capire l'impatto che avessero i nuovi componenti da aggiungere.

Innanzitutto, abbiamo valutato i requisiti funzionali presenti e abbiamo appurato che l'impatto della nostra modifica sarà debole. Le modifiche riguarderanno:

- Modifica del requisito funzionale 2.0, aggiungendo la possibilità di effettuare il login attraverso sistemi esterni.

Dopodiché, abbiamo valutato i casi d'uso. Abbiamo appurato che l'impatto della nostra modifica sarà medio. Esse, consisteranno:

- Aggiunta di un nuovo caso d'uso che mostra il flusso di eventi con l'utilizzo del login mediante l'OAuth2.0

A seguire, abbiamo valutato il database esistente, appurando anche in questo caso che le modifiche avranno un impatto medio. Le modifiche riguarderanno:



- Aggiunta di campi nella tabella degli utenti, per salvarci le informazioni dei vari provider utilizzati dai vari utenti.

Infine, abbiamo analizzato le classi esistenti, considerando un impatto medio della modifica. Infatti, le modifiche riguarderanno la maggior parte del sistema in esistente:

- Modifica nel package `control.gestioneAutenticazione/LoginControl.java` andando ad integrare i vari componenti che verranno utilizzati dall'OAuth 2.0
- Modifica nel package `WebContente/index.jsp` (nella quale è contenuta la pagina di login) andando ad inserire un bottone che consenta di potersi collegare al servizio esterno per realizzare il login mediante l'OAuth 2.0

3.2.3.2 Candidate Impact Set

Classe	Impatto	Numero

3.2.3.3 Discovered Impact Set

Classe	Impatto	Numero

3.2.4 Risultati dell'impact analysis

Per quanto concerne la seguente Change Request, abbiamo effettuato un'analisi delle componenti impattate ma non continueremo con lo sviluppo di essa per motivi di tempistiche legate alla consegna del progetto.



3.3 Change Request 03

Change Request	Data	Richiedente	Priorità
3	29/04/2022	Sviluppatori	Bassa
Descrizione			
Dopo una prima analisi dei riscontri ricevuti dai vari genitori si è deciso di voler ampliare il requisito funzionale 1.0 che riguarda la registrazione di un utente al sistema. Attualmente un utente (genitore) può registrarsi al sistema mediante la sua e-mail, una password e altri campi ma non mediante l'utilizzo di servizi esterni come Google o Facebook.			
Scopo			
L'obiettivo di questa CR è quello di permettere a chi utilizza la web application Summer Center di scegliere al momento della registrazione se effettuarla mediante il suo indirizzo e-mail oppure mediante i servizi di autenticazione di Google o Facebook. Le modifiche riguarderanno l'ampliamento del RF 1.0 il quale riguarda la registrazione al sistema.			
Conseguenza se non accettata			
Il sistema non cambierebbe, ma non consentirebbe l'utilizzo dell'OAuth 2.0 per la registrazione.			

3.3.2 Identificazione, analisi e valutazione

3.3.2.1 Identificazione, descrizione e valutazione dei costi

Identificatore	Valutazione	Motivazione
Integrazione dei requisiti funzionali	Medio	La modifica del requisito 1.0 comporta qualche modifica nelle altre componenti.
Modifica DB	Debole	Aggiunta di qualche campo nel database per salvarci le informazioni del provider dell'OAuth2.0
Implementazione con Java e Javascript	Forte	Lo sviluppatore ha sufficienti conoscenze della tecnologia, ma bisognerebbe implementare tutta l'architettura legata sistema di registrazione.
Testing Funzionale	Forte	Il testing richiede una creazione di tutti i test cases relativi alla registrazione dell'utente alla piattaforma Summer Center.



3.1.2.2 Identificazione, descrizione e valutazione dei benefici

Identificatore	Valutazione	Motivazione
Possibilità di velocizzare l'operazione di registrazione	Forte	Il genitore può velocizzare l'operazione di registrazione attraverso l'utilizzo di sistemi esterni e quindi non deve necessariamente ricordarsi username e password.

3.1.2.3 Identificazione, descrizione e valutazione dei rischi

Identificatore	Valutazione	Motivazione
Introduzione di nuovi fault	Forte	L'attività di testing da effettuare potrà avere un'alta probabilità di riscontrare fault, questo è dovuto al fatto che il sistema non è già stato sottoposto ad una fase di testing.
Carico di lavoro eccessivo per il client	Debole	Il client dovrà solo elaborare dei dati provenienti dal form della registrazione.
Carico di lavoro eccessivo per il server	Debole	Il server dovrà garantire la persistenza dei dati relativi ai diversi utenti registrati al sistema.

Per quanto concerne la seguente Change Request, abbiamo effettuato un'analisi ed una valutazione di tale modifica in termini di tempi e costi ma non continueremo con lo sviluppo di essa per motivi di tempistiche legate alla consegna del progetto.



4. Sistema modificato

4.1 Descrizione del sistema attuale

Il sistema è una web application “Summer Center” volta all’automatizzazione della compilazione dei moduli riguardanti i genitori, i bambini, l’iscrizione e alla gestione della segreteria per risolvere i problemi dell’errata compilazione dei moduli e del sistema di pagamento.

Dopo la modifica, relativa alla CR01, il sistema non fa più riferimento ad un determinato centro, ma ad una catena di centri estivi. Questa modifica ha impattato sia le funzionalità principali del sistema sia i diversi attori del sistema.

L’applicazione offre al genitore le seguenti funzionalità:

- accedere ad un’apposita dashboard dove può visualizzare tutti i dettagli dei propri dati anagrafici, dei bambini e dell’iscrizione;
- compilare gli appositi moduli anagrafici e informativi;
- iscrizione dei bambini ad un determinato centro, scelto tra quelli disponibili, con la possibilità di finalizzare le iscrizioni con un sistema di pagamento online affidabile.
- visualizzare i dettagli delle varie iscrizioni effettuate;
- modificare i propri dati personali;
- decidere di pagare un’iscrizione istantaneamente o successivamente;
- possibilità di richiedere una disdetta di una iscrizione.

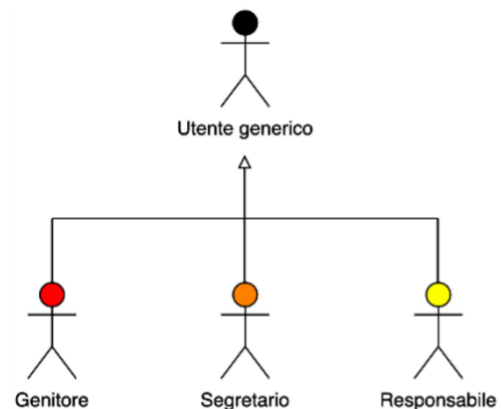
Le funzionalità offerte alla segreteria sono:

- visualizzare tutte le informazioni delle iscrizioni effettuate dai genitori relative al centro di riferimento del segretario;
- scegliere la percentuale di rimborso da attribuire ad un’iscrizione disdetta;
- controllare la validità dell’iscrizione mostrando al sistema il QR Code associato all’iscrizione.

Le funzionalità che il sistema offre al responsabile sono:

- registrare diversi segretari al sistema, assegnano ad ogni segretario un determinato centro estivo;
- visualizzare l’andamento delle iscrizioni dei vari centri estivi attraverso grafici e calcoli statistici;
- registrare un nuovo centro estivo che fa parte della stessa catena di quelli già presenti;
- possibilità di cancellare logicamente un centro estivo.

4.2 Attori del sistema



4.3 Requisiti Funzionali

RF 1 - Registrazione

- **RF 1.0:** Quando il genitore vuole accedere per la prima volta al sistema, Summer Center deve permettere la registrazione dell'account e la verifica dell'account, per poter effettuare successivamente l'accesso al sistema stesso.
- **RF 1.1:** Quando il responsabile vuole attivare un segretario, il sistema deve permettere la registrazione del segretario da parte del responsabile.

RF 2 - Login

- **RF 2.0:** Quando il responsabile o il genitore o il segretario vogliono accedere alle proprie dashboard, il sistema deve permettere l'accesso tramite e-mail e password.

RF 3 - Dashboard

- **RF 3.0:** Il sistema deve permettere la visualizzazione di dashboard e funzionalità diverse in base all'utente loggato.

RF 4 - Genitore

- **RF 4.0:** Il genitore può effettuare un'iscrizione solo dopo aver compilato i propri dati personali obbligatori.
- **RF 4.1:** Il genitore può effettuare il pagamento dell'iscrizione tramite il metodo di pagamento con Carta di credito/Carta di debito o PayPal.
- **RF 4.2:** Il genitore può effettuare la prenotazione selezionando una o più settimane (la settimana va dal lunedì al venerdì) dei mesi compresi tra Maggio e Settembre e può selezionare il centro estivo in cui vuole iscrivere il proprio bambino.
- **RF 4.3:** Il genitore può modificare i propri dati personali.
- **RF 4.4:** Il genitore può visualizzare tutte le iscrizioni da lui effettuate (pagate e non).



- **RF 4.5:** Il genitore può visualizzare i dettagli di una singola iscrizione.
- **RF 4.6:** Il genitore può disdire un'iscrizione.
- **RF 4.7:** Il genitore può cambiare la password per i suoi prossimi accessi.
- **RF 4.8:** Il genitore può decidere di pagare un'iscrizione istantaneamente o successivamente. Decidendo di pagarla successivamente potrà accedere ad una pagina in cui sono presenti le iscrizioni che può selezionare e pagare.

RF 5 – Segretario

- **RF 5.0:** il segretario può visualizzare tutte le iscrizioni effettuate dai genitori per il centro in cui esso è coinvolto il suo centro di appartenenza.
- **RF 5.1:** Il segretario può visualizzare i dettagli di una singola iscrizione.
- **RF 5.2:** Il segretario può scegliere la percentuale di rimborso da attribuire ad un'iscrizione disdetta.
- **RF 5.3:** Il segretario può controllare la validità dell'iscrizione mostrando al sistema il QR Code associato all'iscrizione.
- **RF 5.4:** Il segretario deve obbligatoriamente cambiare password al primo accesso al sistema.
- **RF 5.5:** Il segretario può cambiare la password per i suoi prossimi accessi.

RF 6 – Responsabile

- **RF 6.0:** Il responsabile può registrare segretari al sistema associandoli ad un determinato centro estivo.
- **RF 6.1:** Il responsabile può visualizzare l'andamento delle iscrizioni del centro attraverso grafici e calcoli statistici.
- **RF 6.2:** Il responsabile può cambiare la password per i suoi prossimi accessi.
- **RF 6.3:** Il responsabile può inserire un nuovo centro all'interno del sistema.
- **RF 6.4:** Il responsabile può eliminare logicamente centro all'interno del sistema.

RF 7 – Centro

- **RF 7.0:** ogni centro potrà ospitare al più 100 bambini nello stesso periodo.

4.4 Architettura del sistema

Il sistema attuale, per garantire i servizi, utilizza un'architettura di tipo Client-Server. Il Client è rappresentato dal Browser utilizzato dall'utente.

Come Web Server è utilizzato Apache Tomcat 9.0, dove le pagine JSP/HTML, JS e CSS realizzeranno la UI del sistema e le Java Servlet la logica applicativa. Il client comunica con il Web Server attraverso il protocollo di comunicazione HTTPS.

Il server può comunicare con il database, attraverso le API di JDBC. La gestione dei dati persistenti avviene attraverso il DBMS MySQL.

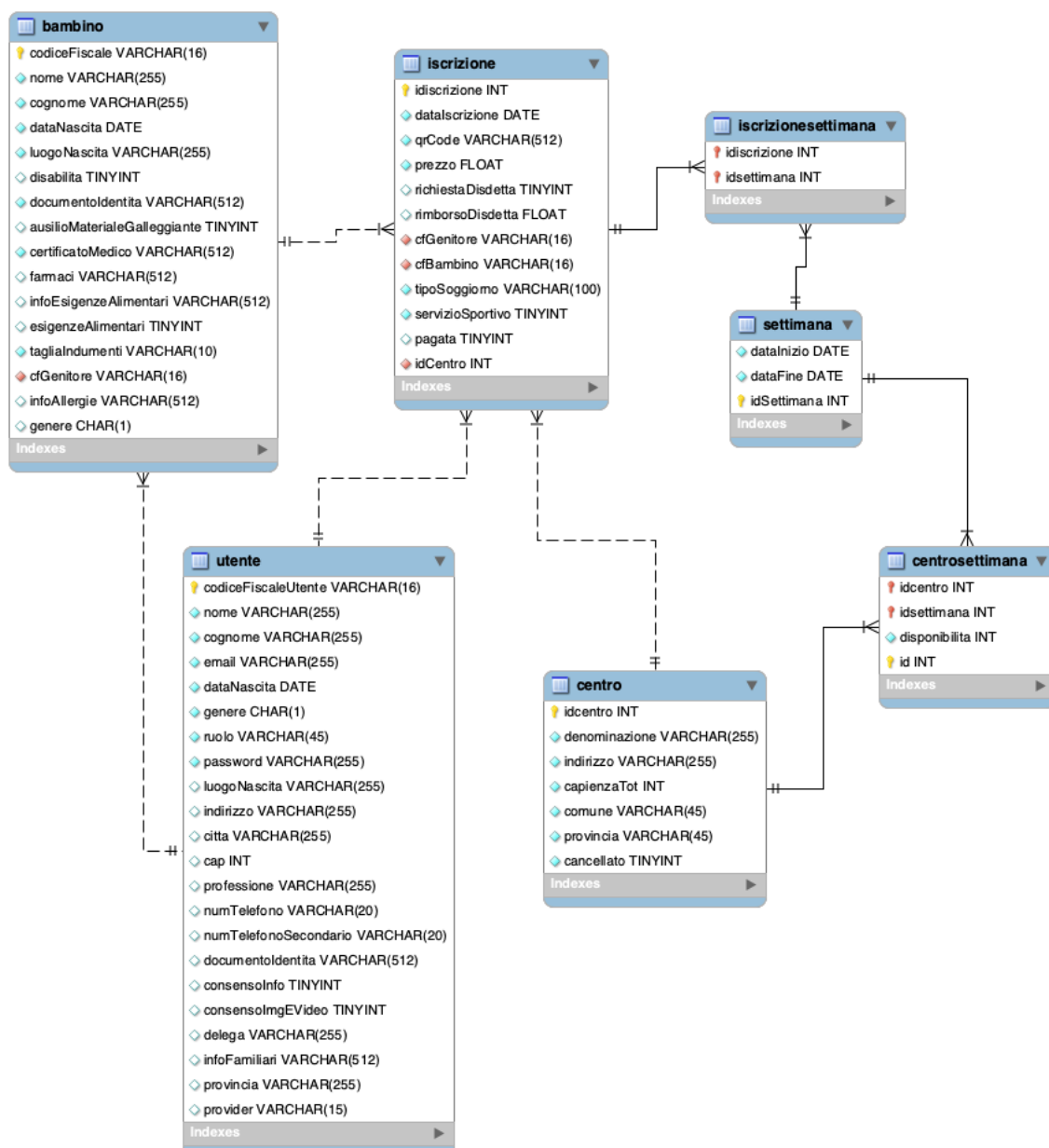


Attraverso il diagramma delle classi abbiamo individuato le entità che saranno memorizzate nel database. Nel nostro caso tutte le entità saranno rese persistenti in un database.

Il controllo globale del software è di tipo Event-Driven Control. Il ciclo principale attende un evento esterno, quando l'evento si verifica viene spedito all'oggetto appropriato. Il controllo risiede in un dispatcher che invoca i servizi di un sottosistema.

Il sistema è implementato, lato client-side mediante il pattern MVC ereditato dal framework utilizzato, lato server-side si necessita di un qualsiasi browser capace di interpretare il linguaggio JavaScript (al fine di permettere l'interazione con l'interfaccia del sistema).

4.4.1 Database



4.4.2 Class Diagram

