

Capitolo 1

Iterazione 0

1.1 Introduzione e panoramica del sistema

Il sistema che verrà implementato in questo progetto di studio si occuperà della gestione di un centro di immersioni, noto anche come *diving center* o *dive center*. Un centro di immersione è una struttura che fornisce supporto, attrezzatura e corsi per la pratica delle attività subacquee. Un centro di immersione fornisce principalmente tre tipi di servizi:

- Immersioni guidate.
- Noleggio attrezzatura.
- Scuola di immersione.

Per immersione guidata si intende un'immersione effettuata in gruppo o singolarmente insieme a guide subacquee addestrate, esperte e a conoscenza dei punti di immersione più importanti ed interessanti. Questa soluzione è ottimale quando ci si vuole immergere in luoghi non conosciuti o in parchi marini protetti per cui è necessaria un'autorizzazione e una guida. Inoltre il centro diving offre i servizi di accompagnamento e assistenza, di trasposto con imbarcazioni o gommoni, che permettono l'entrata e l'uscita dall'acqua con praticità e comfort.

Il focus del nostro progetto sarà la gestione e l'organizzazione efficiente delle immersioni guidate e la gestione del noleggio attrezzatura, non verrà incluso l'insegnamento della pratica subacquea.

Un centro di immersione ha una disponibilità limitata di barche e guide per cui ha la possibilità di effettuare un numero limitato di escursioni durante una giornata. Il sistema che verrà sviluppato si occuperà di organizzare al meglio l'allocazione delle risorse disponibili affinché un numero maggiore di persone possa parteciparvi. Spesso le persone che vogliono partecipare a questo tipo di escursioni si trovano in gruppo, per cui è necessario ottimizzare l'allocazione delle risorse disponibili, che in questo caso costituiscono i posti disponibili in barca, con il vincolo di mantenere intatti i gruppi.

Il sistema implementato rappresenterà un centro di immersione: si occuperà della gestione delle risorse (barche, escursioni, attrezzatura disponibile) e della gestione delle prenotazioni delle escursioni da parte degli utenti.

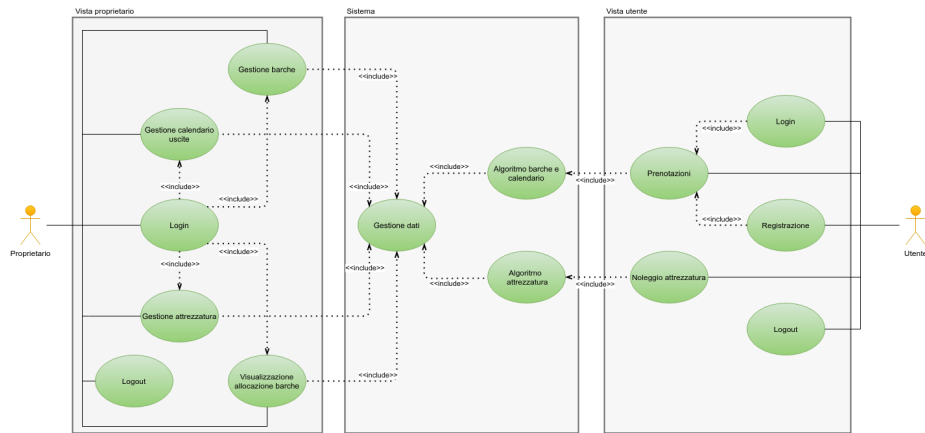


Figura 1.1: Diagramma UML dei casi d'uso

1.2 Requisiti funzionali e analisi dei casi d'uso

In questa sezione verranno introdotti i requisiti funzionali del sistema attraverso l'utilizzo dei casi d'uso.

Lo schema UML dei casi d'uso viene riportato in Figura 1.1.

1.3 Requisiti non funzionali

Il progetto verrà sviluppato tenendo in considerazione anche alcuni requisiti non funzionali, quali la manutenibilità, l'efficienza e l'usabilità. Il requisito di manutenibilità verrà rispettato mediante l'implementazione di tutte le risorse persistenti ovvero barche, orari e date escursioni, attrezzatura. In questo modo anche se dovessero cambiare nel futuro alcune condizioni sarà facile per il proprietario rifletterle all'interno del software. Il requisito dell'efficienza è l'obiettivo del progetto, ovvero l'allocazione ottimale delle risorse disponibili, i posti presenti nelle barche. L'usabilità è garantita dalla decisione di sviluppare il programma attraverso un'applicazione Android, di facile utilizzo sia per gli utenti che per il proprietario. In base alla topologia (Figura 1.2) è possibile notare che il database verrà ospitato da un hosting online e quindi sarà sempre possibile, mediante cellulare, accedere a tutte le funzionalità dell'applicazione.

1.4 Topologia del sistema

La topologia del sistema mostrata in Figura 1.2 evidenzia come sia gli utenti che il proprietario accedono alla piattaforma tramite app Android.

Questa mette a disposizione un set di API REST con cui è possibile accedere alle funzionalità dell'applicazione che salverà i dati su un database online.

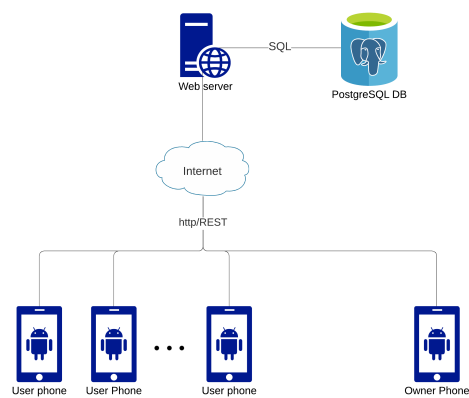


Figura 1.2: Topologia del sistema.

Capitolo 2

Iterazione 1

2.1 Introduzione

Nella prima iterazione si è scelto di implementare i seguenti casi d'uso:

- UC1: Gestione barche [*Astratto*]
 - UC1.1 Visualizzazione barche
 - UC1.2 Inserimento barca
 - UC1.3 Modifica barca
 - UC1.4 Eliminazione barca
- UC2: Gestione calendario uscite [*Astratto*]
 - UC2.1 Visualizzazione uscite
 - UC2.2 Inserimento uscita
 - UC2.3 Modifica uscita
 - UC2.4 Eliminazione uscita
- UC3: Registrazione utente
- UC4: Login utente

Dopo averli descritti in maniera testuale si è passati alla realizzazione del diagramma dei componenti e del deployment diagram. Infine il codice è stato implementato sulla base di questi documenti.

2.2 UC1: Gestione barche

Breve descrizione: il proprietario del diving center deve avere una visione generale e un controllo completo delle sue barche all'interno dell'applicazione. Oltre alla visualizzazione, il proprietario deve poter modificare ed eliminare le imbarcazioni già presenti nel sistema e poter aggiungerne di nuove. La gestione delle barche è quindi suddivisa in 4 casi d'uso concreti:

- UC1.1 Visualizzazione barche

- UC1.2 Inserimento barca
- UC1.3 Modifica barca
- UC1.4 Eliminazione barca

Attori coinvolti: Proprietario, Sistema.

Trigger: login del proprietario avvenuto con successo.

Postcondizione: il proprietario dopo il login, viene indirizzato alla home page e può scegliere se cliccare uno dei button: *Visualizza barche* o *Aggiungi barca*. Il primo per visualizzare tutte le barche inserite, il secondo per aggiungerne una nuova.

Procedimento:

1. Proprietario effettua il login.
2. Sistema mostra la pagina iniziale che comprende un button *Visualizza barche* e un button *Aggiungi barca* con cui il proprietario può concretamente gestire le sue barche.

2.2.1 UC1.1 Visualizzazione barche

Breve descrizione: Il proprietario visualizza le barche inserite nel sistema e le relative caratteristiche.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button *Visualizza barche*.

Postcondizione: Il sistema mostra una vista con l'elenco delle barche.

Procedimento:

1. Proprietario effettua il login
2. Proprietario clicca il button *Visualizza barche* presente nella home page.
3. Sistema mostra l'elenco delle barche presenti nel database con le seguenti informazioni:
 - Nome della barca
 - Modello della barca
 - Numero posti disponibili sulla barca
 - Altre caratteristiche TODO
4. Per ogni item dell'elenco il proprietario può cliccare su *Modifica barca* o *Elimina barca* per modificare o eliminare una barca.

2.2.2 UC1.2 Inserimento barca

Breve descrizione: Il proprietario aggiunge una barca nel sistema. Questo viene fatto principalmente nella fase iniziale in cui il proprietario deve inserire tutte le sue barche all'interno dell'applicazione e poi in seguito all'acquisto di nuove barche.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button *Inserisci barca*.

Postcondizione: Il sistema mostra la nuova barca all'interno della pagina di visualizzazione delle barche.

Procedimento:

1. Il proprietario si trova sulla home page e clicca il button *Inserisci barca*.
2. Sistema mostra un form in cui è possibile inserire i dati relativi alla barca.
3. Proprietario riempie il form.
4. Proprietario clicca su *Conferma* per inserire la barca nel sistema.
5. Il sistema mostra la nuova barca all'interno della pagina di visualizzazione delle barche.

2.2.3 UC1.3 Modifica barca

Breve descrizione: Il proprietario si trova sulla pagina di visualizzazione delle barche e clicca su un button o un'icona di modifica di una delle barche. Tale modifica può essere dovuta a cambiamenti strutturali della barca, per esempio la riduzione del numero di posti disponibili.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button o l'icona *Modifica barca*.

Postcondizione: Il sistema mostra la barca modificata all'interno della pagina di visualizzazione delle barche.

Procedimento:

1. Il proprietario si trova sulla pagina di visualizzazione delle barche e clicca su un button o un'icona di modifica di una delle barche.
2. Sistema mostra una vista in cui è possibile modificare i dati relativi alla barca.
3. Proprietario modifica i dati.
4. Proprietario clicca su *Conferma* per modificare i dati della barca.
5. Il sistema mostra la barca aggiornata all'interno della pagina di visualizzazione delle barche.

2.2.4 UC1.4 Eliminazione barca

Breve descrizione: Il proprietario si trova sulla pagina di visualizzazione delle barche e clicca su un button o un'icona di eliminazione di una delle barche. L'eliminazione può essere definitiva o temporanea. L'eliminazione è definitiva se la barca non è più agibile o perché viene sostituita da un'altra; è temporanea nel caso in cui sia necessaria attività di manutenzione.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button o l'icona *Elimina barca*.

Postcondizione: Il sistema mostra la pagina di visualizzazione delle barche in cui non comparirà la barca eliminata.

Procedimento:

1. Il proprietario si trova sulla pagina di visualizzazione delle barche e clicca su un button o un'icona di eliminazione di una delle barche. Il proprietario deve inoltre scegliere se l'eliminazione è temporanea o definitiva.
2. Sistema mostra un alert che avvisa il proprietario che l'azione è irreversibile.
3. Il proprietario può scegliere se confermare l'eliminazione cliccando su *Conferma* o annullare l'azione cliccando su *Annulla*.
4. Il sistema mostra la pagina di visualizzazione delle barche in cui non comparirà la barca eliminata se il proprietario ha scelto di eliminarla in maniera non temporanea. Se invece, la barca è stata eliminata temporaneamente, essa sarà sempre visibile nella pagina di visualizzazione delle barche, ma segnalando al proprietario che il sistema organizzerà le uscite senza tenere conto di quella barca.

2.3 UC2: Gestione calendario uscite

L'applicazione deve poter semplificare la gestione e l'organizzazione delle uscite al proprietario del diving center. L'applicazione deve fungere da calendario, mettendo a disposizione le seguenti funzionalità:

- UC2.1 Visualizzazione uscite
- UC2.2 Inserimento uscita
- UC2.3 Modifica uscita
- UC2.4 Eliminazione uscita

2.3.1 UC2.1 Visualizzazione uscite

Breve descrizione: Il proprietario visualizza le date e i turni in cui gli utenti possono prenotarsi.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button *Visualizza uscite*.

Postcondizione: Il sistema mostra una vista con l'elenco delle uscite.

Procedimento:

1. Proprietario effettua il login.
2. Proprietario clicca il button *Visualizza uscite* presente nella home page.
3. Sistema mostra l'elenco delle uscite rese disponibili con le seguenti informazioni:
 - Data dell'uscita.
 - Orario di inizio uscita.
 - Orario di fine uscita.
 - Altro TODO
4. Per ogni data presente nell'elenco il proprietario può cliccare su *Modifica uscita* o *Elimina uscita* per modificare o eliminare un'uscita.

2.3.2 UC2.2 Inserimento uscita

Breve descrizione: Il proprietario aggiunge un'uscita al calendario indicandone data e orari in cui verranno rese disponibili le prenotazioni.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button *Inserisci uscita*.

Postcondizione: Il sistema mostra la nuova uscita all'interno della pagina di visualizzazione delle uscite.

Procedimento:

1. Il proprietario si trova sulla home page e clicca il button *Inserisci uscita*.
2. Sistema mostra un form in cui è possibile inserire i dati relativi all'uscita.
3. Proprietario riempie il form.
4. Proprietario clicca su *Conferma* per inserire l'uscita nel sistema.
5. Il sistema mostra la nuova uscita all'interno della pagina di visualizzazione delle uscite.

2.3.3 UC2.3 Modifica uscita

Breve descrizione: Il proprietario si trova sulla pagina di visualizzazione delle uscite e clicca su un button o un'icona di modifica di una delle uscite. Tale modifica può essere dovuta a cambiamenti climatici o ad altri imprevisti.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button o l'icona *Modifica uscita*.

Postcondizione: Il sistema mostra l'uscita modificata all'interno della pagina di visualizzazione delle uscite.

Procedimento:

1. Il proprietario si trova sulla pagina di visualizzazione delle uscite e clicca su un button o un'icona di modifica di una delle uscite.
2. Sistema mostra una vista in cui è possibile modificare i dati relativi all'uscita.
3. Proprietario modifica i dati.
4. Proprietario clicca su *Conferma* per modificare i dati dell'uscita.
5. Il sistema mostra l'uscita aggiornata all'interno della pagina di visualizzazione delle uscite.

2.3.4 UC2.4 Eliminazione uscita

Breve descrizione: Il proprietario si trova sulla pagina di visualizzazione delle uscite e clicca su un button o un'icona di eliminazione di una delle uscite.

Attori coinvolti: Proprietario, Sistema.

Trigger: Il proprietario clicca il button o l'icona *Elimina uscita*.

Postcondizione: Il sistema mostra la pagina di visualizzazione delle uscite in cui non comparirà l'uscita eliminata.

Procedimento:

1. Il proprietario si trova sulla pagina di visualizzazione delle uscite e clicca su un button o un'icona di eliminazione di una delle uscite.
2. Sistema mostra un alert che avvisa il proprietario che l'azione è irreversibile.
3. Il proprietario può scegliere se confermare l'eliminazione cliccando su *Conferma* o annullare l'azione cliccando su *Annulla*.
4. Il sistema mostra la pagina di visualizzazione delle uscite in cui non comparirà l'uscita eliminata.

2.4 UC3: Registrazione dell'utente

Breve descrizione: L'utente compila il form per la registrazione all'app e, se non si è già registrato, viene aggiunto al database.

Attori coinvolti: Utente, Sistema?

Trigger: L'utente preme su "Registrazione".

Postcondizione: L'utente è stato inserito nel database e ha ricevuto la conferma dell'operazione.

Procedimento:

1. Utente preme su "Registrazione" nella pagina iniziale dell'app

2. Utente fornisce Nome, username, indirizzo email e password nel form di registrazione
3. Il sistema verifica se esiste già un utente con quella mail:
 - (a) se esiste già, il sistema lo comunica all'utente
 - (b) se non esiste allora il sistema aggiunge l'utente nel database

2.5 UC4: Login dell'utente

Breve descrizione: L'utente (utente normale o il proprietario) compila il form per il login e se le credenziali sono giuste il sistema gli consente l'accesso.

Attori coinvolti: Utente/Proprietario, Sistema?

Trigger: L'utente preme su "Login"

Postcondizione: L'utente ha accesso alla vista del suo profilo.

Procedimento:

1. Utente preme su "Login" nella pagina iniziale dell'app
2. Utente fornisce username e password nel form di registrazione
3. Il sistema controlla le credenziali inserite:
 - (a) se sono corrette, il sistema invia una conferma di accesso all'utente
 - (b) se sono errate, il sistema lo comunica all'utente

2.6 UML Component diagram

I casi d'uso scelti in questa prima iterazione vengono rappresentati sottoforma di componenti nel diagramma in Figura 2.1. Si è scelto di suddividere i caso d'uso in componenti:

- «boundary» rappresentati dai componenti lato front-end con cui gli attori si interfacciano direttamente. Tali componenti richiedono delle interfacce al back-end.
- «control» rappresentati dai componenti lato back-end che forniscono delle API al front-end, richiedonone a loro volta al database.
- «data» rappresentato dal database in cui verranno memorizzati i dati delle barche, il calendario delle uscite e i dati dell'utente in occasione della registrazione.

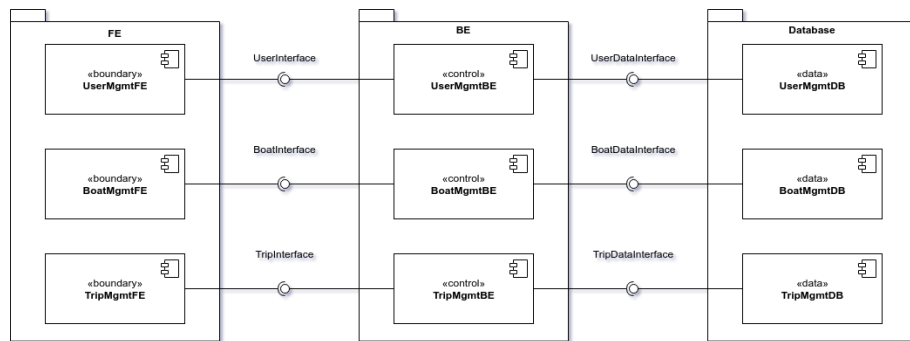


Figura 2.1: Diagramma dei componenti UML.

2.7 UML Class diagram per Interfacce

Il diagramma in Figura 2.2 serve a rappresentare le interfacce del sistema e le loro interazioni.

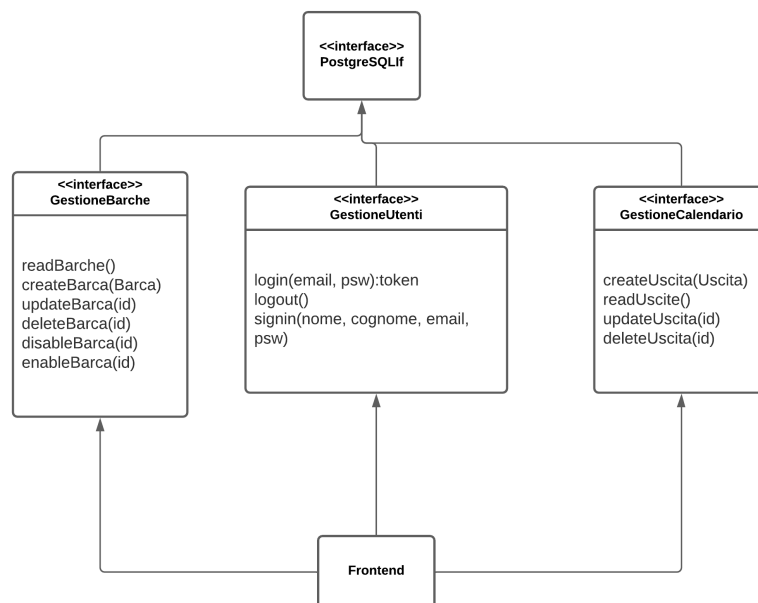


Figura 2.2: Diagramma dei componenti UML.

2.8 UML Class diagram per tipi di dato

Il seguente diagramma serve ad esplicitare tipi di dato particolari e i loro legami.

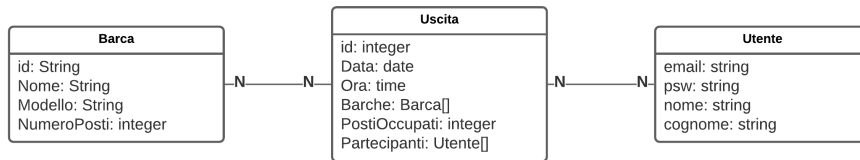


Figura 2.3: Diagramma dei componenti UML.

2.9 UML Deployment diagram

I componenti descritti precedentemente vengono istanziati nel Deployment diagram. In Figura 2.4 vengono mostrati i componenti contenuti nei seguenti nodi:

- Cellulare proprietario è il nodo su cui l'admin del sistema potrà gestire le barche e il calendario delle uscite.
- Cellulare utente è il nodo su cui una persona potrà registrarsi alla piattaforma.
- Web server fornisce le API richieste dall'applicativo lato front-end.
- Database funge da storage dei dati.

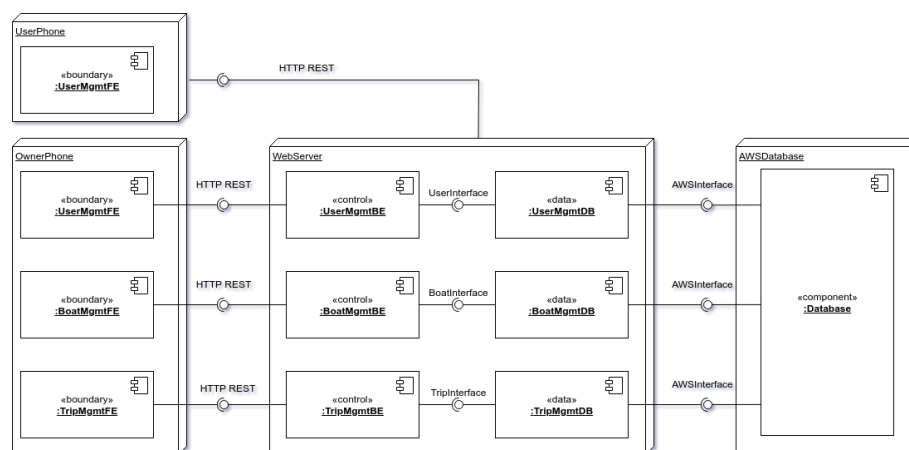


Figura 2.4: Deployment diagram UML.