**Elaborato per il corso di Basi di dati**

**A.A 2023/2024**

Università di Bologna - Ingegneria e Scienze Informatiche

**Progetto di una base di dati per la gestione di missioni spaziali**



Figura 1 Logo ufficiale Aurora Rocketry Team - Unibo

Luca Pulga

luca.pulga@studio.unibo.it

000191461

Analisi dei requisiti

Abstract

Intervista

Azioni richieste

Prospettiva futura

Estrazione concetti principali

Progettazione concettuale

Schema scheletro

Schema finale

Progettazione logica

Stima del volume dei dati

Descrizione delle operazioni principali e stima della loro frequenza

Schemi di navigazione e tabelle degli accessi

Raffinamento dello schema

Analisi delle ridondanze

Traduzione di entità e associazioni in relazioni

Schema relazionale finale

Traduzione delle operazioni in query SQL

Progettazione dell’applicazione

Descrizione dell’architettura dell’applicazione realizzata

# **Analisi dei requisiti**

## Abstract

L’obiettivo del progetto è realizzare un sito web con relativo database che possa permettere il salvataggio delle missioni spaziali a cura del team universitario dell’Università di Bologna “Aurora Rocketry Team” in vista della competizione “Euroc”, European Rocketry Competition.

Il sito fornirà, agli studenti del team, un'interfaccia centralizzata per pianificare, monitorare e analizzare le missioni spaziali, inclusi lanci di razzi, esperimenti scientifici e raccolta di dati.

Sarà disponibile una pagina dedicata alla registrazione degli utenti, dove ogni membro del team potrà creare un account inserendo le proprie informazioni personali e di contatto.

Saranno implementate operazioni come l’inserimento di missioni, in cui gli utenti possono inserire i dettagli di una nuova missione, compresi obiettivi, data di lancio prevista, equipaggiamento necessario e requisiti tecnici. La modifica e l’eliminazione di missioni esistenti, in cui gli utenti autorizzati possono apportare modifiche ai dettagli delle missioni esistenti o eliminarle se necessario.

Sarà possibile per ogni missione pianificare e visualizzare i lanci di razzi con relativi sensori, payload e sponsor che aiutano nella progettazione dei vari aspetti tecnici e non.

In più sarà possibile la visualizzazione di informazioni statistiche complesse, in quanto gli utenti avranno accesso a dati statistici dettagliati sulle performance delle missioni passate, inclusi grafici sull'efficienza del lancio, il successo degli obiettivi raggiunti e altre metriche rilevanti per il miglioramento delle prestazioni future. Sarà implementata una gestione degli utenti e dei permessi.

## Intervista

Un primo testo ottenuto dall’intervista è il seguente:

Il progetto prevede la creazione di un sistema per tenere traccia delle missioni spaziali eseguite dal team. Ogni missione sarà identificata da un nome univoco e avrà uno stato che ne descrive il progresso, potendo essere aperta, pianificata, in corso, completata o annullata. Sarà inoltre associata a una data di lancio e potrà contenere note aggiuntive per registrare informazioni rilevanti. Gli obiettivi della missione, univoci per ciascuna missione, definiranno gli scopi specifici di ogni operazione.

Il luogo di lancio, elemento fondamentale per ogni missione, sarà salvato nel database, così come il payload, ovvero il carico scientifico trasportato dal razzo. Ogni missione avrà un unico payload, caratterizzato da un ID, un nome, un peso e, se necessario, dimensioni.

Le missioni potranno essere supportate da sponsor, i quali potranno contribuire sia con materiale e consulenze tecniche, sia con supporto finanziario.

Il razzo utilizzato per ogni missione sarà unico e identificato da un nome, un peso e delle dimensioni specifiche. Il motore del razzo, elemento cruciale per la sua funzionalità, avrà una serie di parametri tracciati nel sistema.

A bordo del razzo saranno montati vari sensori, ciascuno con un nome, una tipologia specifica, il produttore, l'unità di misura, l'accuratezza e la frequenza di campionamento. Inoltre, ogni razzo sarà equipaggiato con un paracadute per il ritorno a terra, di cui verranno registrati il nome, il produttore, il tipo e il diametro.

Per la strumentazione e il materiale utilizzato nelle missioni, sarà possibile associare dei link, che potranno fornire riferimenti utili relativi a specifici componenti.

Per ogni razzo andranno svolte delle simulazioni a cura dei vari dipartimenti del team. Le simulazioni dunque saranno identificate da un id, un nome e dalla tipologia.

Per quanto riguarda l’organigramma del team, si vuole poterlo suddividere e salvare in dipartimenti. In ogni dipartimento possono coesistere più sub-team. Per ogni dipartimento si vuole salvare un capo dipartimento e stessa cosa per ogni sub-team. Dei team-leaders si vuole mantenere lo storico.

Inoltre si vuole poter monitorare le azioni che ogni utente svolge sull’applicativo attraverso i log.

Infine, per garantire il monitoraggio e la sicurezza delle operazioni, si vuole poter tracciare le azioni che ogni utente svolge sull’applicativo attraverso dei log, registrando tutte le attività compiute per avere una cronologia dettagliata delle operazioni eseguite.

## Estrazione concetti principali

|  |  |  |
| --- | --- | --- |
| Termine | Breve descrizione | Eventuale sinonimo |
| Team Leader | Team Leader di un team all’interno di un dipartimento | Team Leader |
| Capo Dipartimento | Capo di un dipartimento, il quale comprende vari team | Department Leader |
| Payload | Strumento/i o un qualsiasi carico montato sul razzo | - |

A seguito della lettura e comprensione dei requisiti, si procede redigendo un testo che ne riassuma tutti i concetti e in particolare ne estragga quelli principali eliminando le ambiguità sopra rilevate:

Si vuole tenere traccia delle **missioni** **spaziali** eseguite dal team, le quali disporranno di un nome, uno stato (aperta, pianificata, in corso, completata, annullata), la data di lancio ed ulteriori o eventuali note.

Ogni **missione** può essere utilizzata per vari scopi ed **obiettivi**, i quali obiettivi sono univoci per ogni missione. Ogni missione ha come **luogo** di lancio un determinato luogo, anch’esso da salvare specificando stato, regione, provincia, città, via. Inoltre per ogni missione verrà deciso un **payload**, ovvero un carico che verrà montato sul razzo per motivi e scopi scientifici di altro genere. Il payload sarà costituito da un id, un nome, un peso ed eventuali opzionali dimensioni. In una missione sarà possibile caricare un e uno solo payload.

Ogni missione può avere a supporto degli **sponsor**. Gli sponsor possono fornire supporto economico/finanziario nel caso di uno **sponsor** **finanziario**, in cui andrà specificato il finanziamento, possono fornire consulenze se di tipo **consulenziale**, specificando di che tipo, oppure possono essere di tipo **fornitore**, in caso forniscano al team materiale, strumenti o altri supporti per la missione.

Per ogni missione verrà lanciato un solo **razzo**, il quale sarà identificato da un nome, da un peso e le sue dimensioni. Inoltre sarà costituito da un motore del quale si vuole tener traccia avrà una serie di parametri tracciati nel sistema, tra cui il nome, il produttore, la spinta massima, il tipo di carburante utilizzato, la massa, la durata della combustione e l'impulso specifico.

Ogni razzo sarà costituito da più **sensori** dei quali si vuol tener traccia del loro nome, della tipologia (come accelerometro, giroscopio, termometro, attuatore, ecc.), il produttore, dell’unità di misura, accuratezza e frequenza di campionamento e stessa cosa per i/il microcontrollori/e montati sul razzo.

Ogni razzo avrà a disposizione un **paracadute** per il ritorno a terra. Di esso si vuole salvare il nome, il produttore, il tipo e il diametro.

Di tutta la strumentazione e materiale utilizzato si vuole poter traccia anche di eventuali link, ad esempio vorrei poter salvare dei link relativi ad un determinato sensore salvato del database.

Ogni razzo sarà inoltre soggetto a **simulazioni** condotte dai vari **dipartimenti** del team. Queste simulazioni saranno identificate da un ID, un nome e una tipologia, consentendo di monitorare i test e le prove effettuate.

L’organizzazione del team verrà suddivisa in dipartimenti, ciascuno dei quali potrà includere più **team**. Ogni dipartimento avrà un **capo-dipartimento**, e ogni sub-team avrà un **team-leader**. Sarà importante mantenere uno storico dei capi-dipartimento e dei team-leaders per tenere traccia dei cambiamenti nella leadership e dei vari cambiamenti di membri all’interno dei team.

## Azioni richieste

Segue un elenco delle principali azioni richieste:

* Inserimento di tutte le entità riportate nello schema logico
* Aggiornamento di tutte le entità riportate
* Eliminazione di tutte le entità riportate
* Selezionare gli obiettivi delle relative missioni
* Selezionare i sensori associati ai vari razzi
* Selezione dei dati rilevati dai sensori associati ad una missione raggruppati per sensore
* Seleziona le missioni associate alle aziende fornitrici (analogo per le aziende di consulenza e finanziatrici)

## Prospettiva futura

In aggiunta alle funzionalità già descritte, il sito web in futuro, per scopi legati alla gara studentesca, sarà in grado di ricevere e memorizzare dati provenienti da sensori di vario tipo, quali accelerometri, giroscopi, sensori di pressione e temperatura, installati sul razzo durante la missione. Sarà dunque possibile per gli utenti monitorare i dati in tempo reale attraverso grafici e visualizzazioni interattive direttamente sull'interfaccia del sito web durante il lancio e la fase di volo.

Inoltre verrà implementata un’analisi dei dati post-missione più dettagliata e complessa. Gli utenti potranno esaminare i dati dettagliati, confrontarli con gli obiettivi pianificati e trarre conclusioni per migliorare le future missioni.

## **Progettazione concettuale**

## Missioni

### Progettazione dello schema E/R

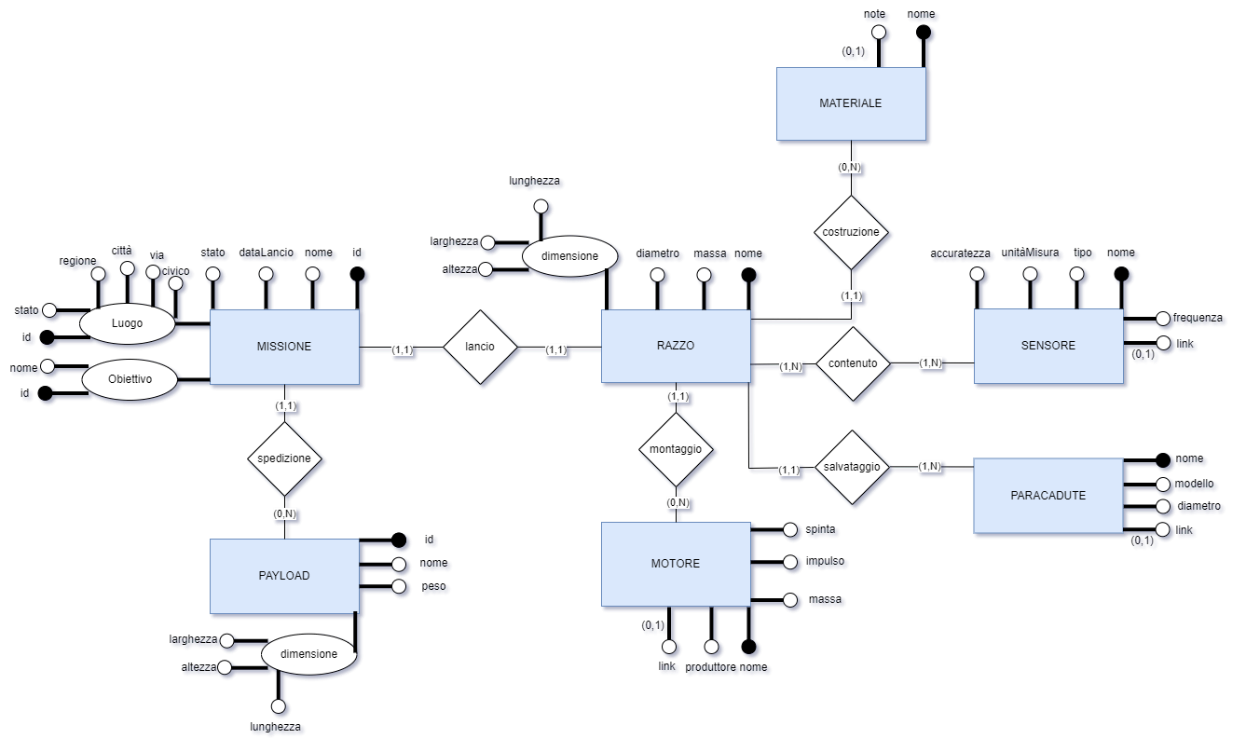
Dopo aver esaminato il dominio del problema per quanto riguarda la gestione delle missioni insieme alla gestione dei luoghi, degli obiettivi dei payload e dei sensori utilizzati sui vari razzi, viene proposto il seguente schema scheletro:

Figura 2 Modello E/R iniziale per missioni, luoghi, payload, motori, materiali, paracaduti e sensori

### Raffinamenti proposti

Una missione può essere compiuta in un solo luogo, i quali luoghi di lancio verranno salvati in una tabella a sé stante. Stesso procedimento per gli obiettivi, i quali possono a loro volta essere realizzati in diverse missioni. Si ipotizza che un motore montato su un razzo possa essere riutilizzato, dunque che non sia monouso, oppure che non sia ancora stato utilizzato e in un razzo si può montare solamente un motore.

Si ipotizza che in una missione possa essere riutilizzato un e un solo razzo. Verrà utilizzato un razzo per ogni nuova missione, i sensori utilizzati possono essere riutilizzati per ogni altra nuova missione, sempre che questi siano integri e funzionanti. In ogni missione è possibile spedire un solo payload e ogni payload può essere riutilizzato per più missioni. Si assume che inizialmente un razzo sia costituito principalmente di un solo tipo di materiale (es carbonio, fibra di vetro, etc). Si assume che un paracadute, per riportare a terra il razzo, possa essere utilizzato per più razzi e dunque più missioni e che un razzo possa ospitare solamente un paracadute.

### Schema concettuale parziale

Dopo l’applicazione dei raffinamenti proposti e l’inserimento degli attributi necessari per le varie entità, lo schema E/R parziale risulta essere di questo tipo:

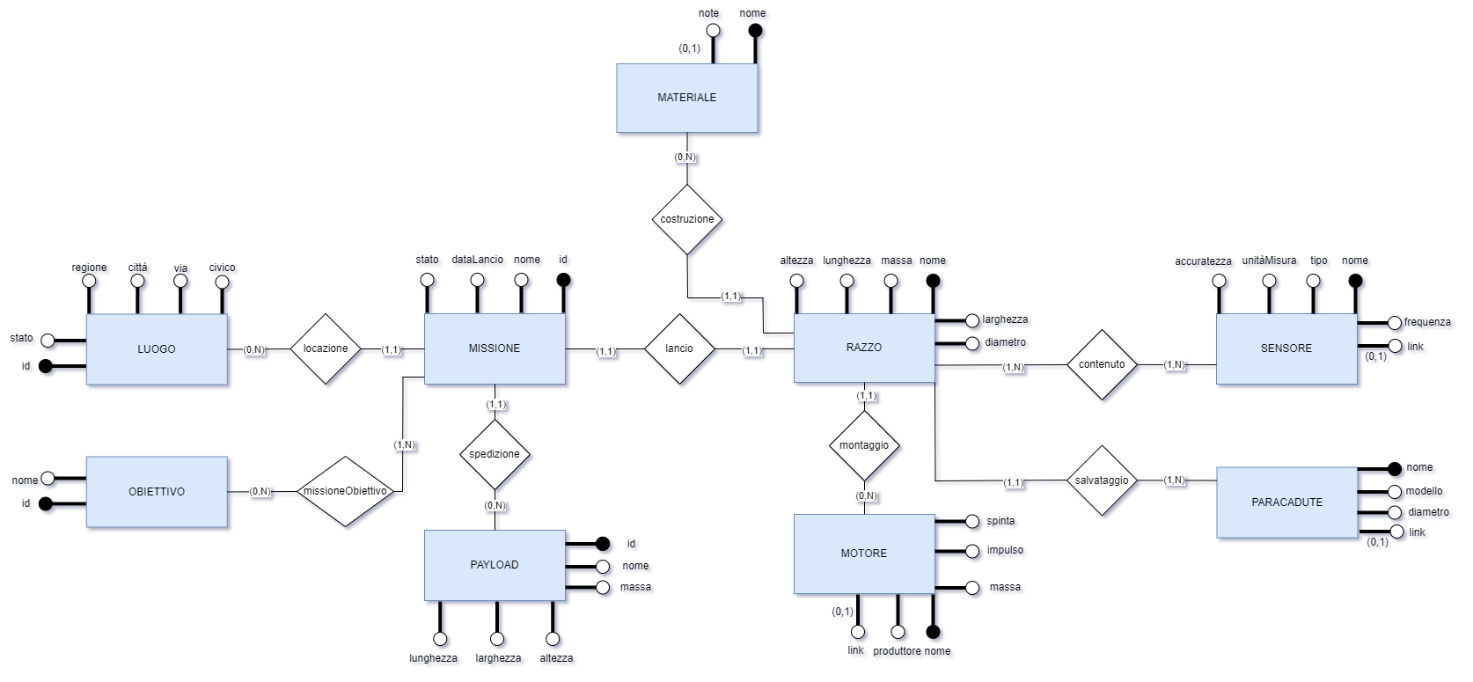


Figura 3 Modello E/R parziale per missioni, luoghi, payload, motori, materiali, paracaduti e sensori

## Aziende e Missioni

### Progettazione dello schema E/R

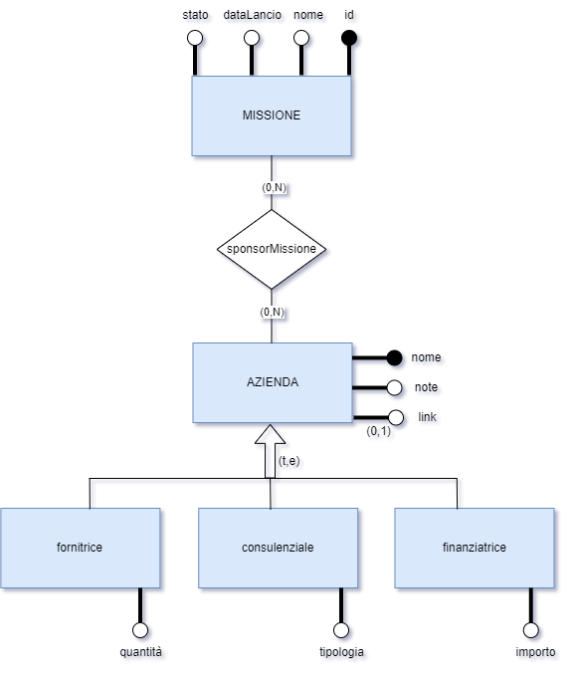
Dopo aver esaminato il dominio del problema per quanto riguarda la gestione delle missioni e delle aziende sponsor, le quali possono fornire varie tipologie di servizi e benefici come nella gerarchia qui presentata, viene proposto il seguente schema scheletro:

Figura 4 Modello E/R iniziale per aziende e missioni

### Raffinamenti proposti

Un'azienda può sponsorizzare N missioni e una missione può essere sponsorizzata da nessuna o da più aziende, in quanto inizialmente sarà difficile ottenere sponsorship dunque si ipotizza che alcune missioni possano non avere alcun sponsor. Per quanto riguarda la gerarchia delle tipologie di aziende che possono offrire benefici, si è deciso di eliminare l'entità madre "azienda" e far coesistere le entità figlie in quanto è necessario mantenere i vari parametri relativi ad ogni entità.

### Schema concettuale parziale

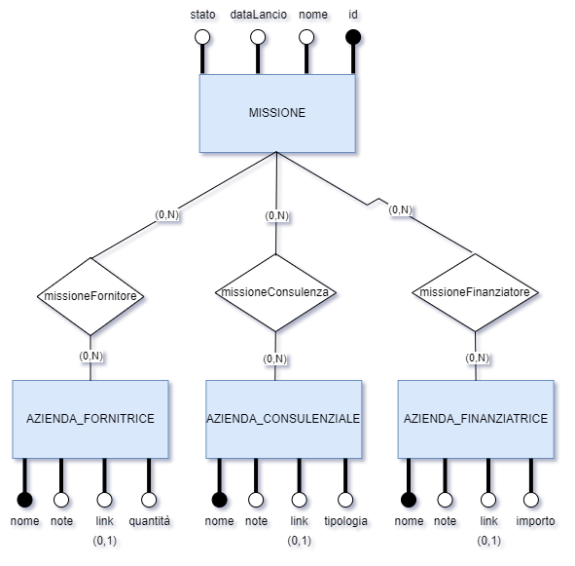
Dopo l’applicazione dei raffinamenti proposti e l’inserimento degli attributi necessari per le varie entità, lo schema E/R parziale risulta essere di questo tipo:

Figura 5 Modello E/R parziale per aziende e missioni

## Organigramma

### Progettazione dello schema E/R

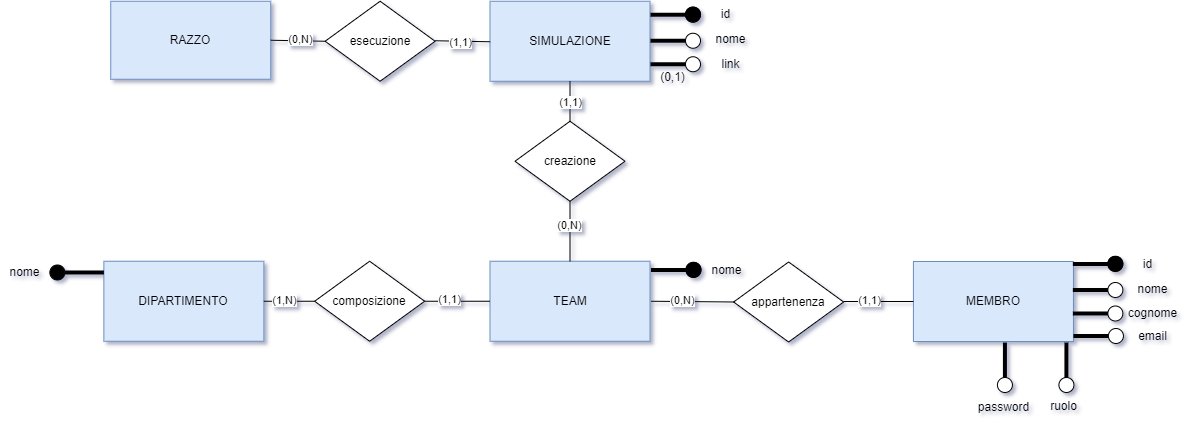
Dopo aver esaminato il dominio del problema per quanto riguarda la gestione dell’organigramma dell’Aurora Rocketry Team, viene proposto il seguente schema scheletro:

Figura 6 Modello E/R iniziale per l'organigramma del team

### Raffinamenti proposti

I membri vengono gestiti tramite un id univoco, evitando di utilizzare il codice fiscale per evitare problemi di omocodia. Inoltre è necessario mantenere anche uno storico delle appartenenze ai team e dei ruoli dei membri del personale nel tempo, in modo che ogni cambiamento di stato sia registrato e tracciato con precisione. La soluzione proposta successivamente permette di mantenere uno storico completo dei movimenti di ciascun dipendente all'interno dell'organizzazione, assicurando che un dipendente sia attualmente assegnato a un solo team alla volta e sia possibile gestire le promozioni o cambiamenti di ruolo nel tempo. In aggiunta ogni team sarà responsabile di effettuare le relative simulazioni per ogni razzo presentato durante le missioni.

### Schema concettuale parziale

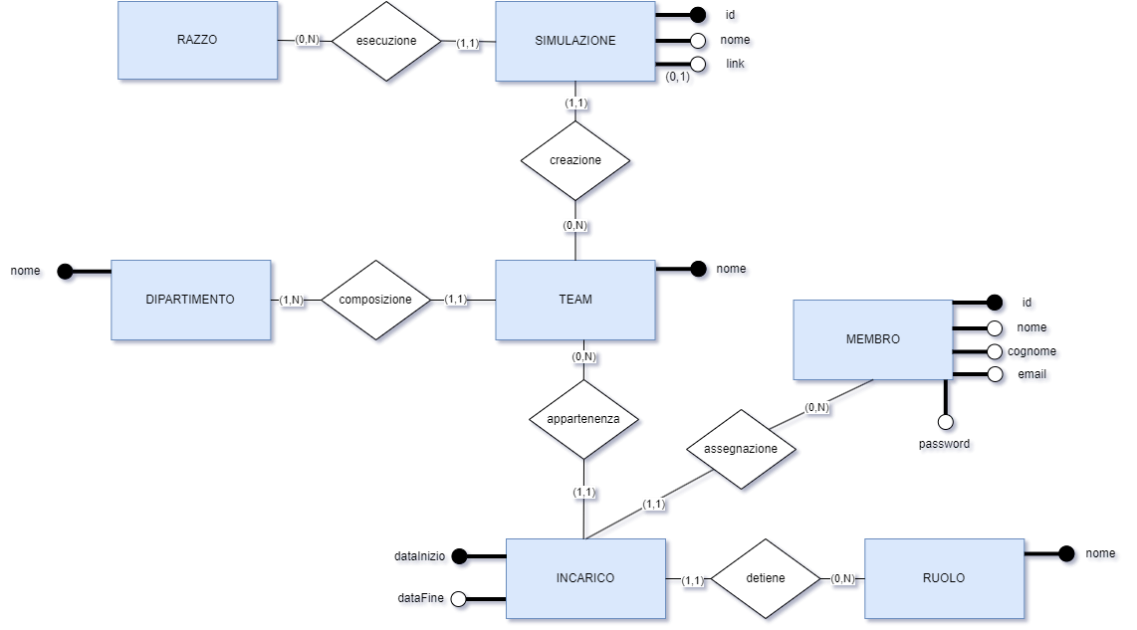
Dopo l’applicazione dei raffinamenti proposti e l’inserimento degli attributi necessari per le varie entità, lo schema E/R parziale risulta essere di questo tipo:

Figura 7 Modello E/R parziale per l'organigramma del team

## Misurazione dei dati

### Progettazione dello schema E/R

Dopo aver esaminato il dominio del problema per quanto riguarda la gestione della misurazione dei dati provenienti dai sensori, viene proposto il seguente schema scheletro:

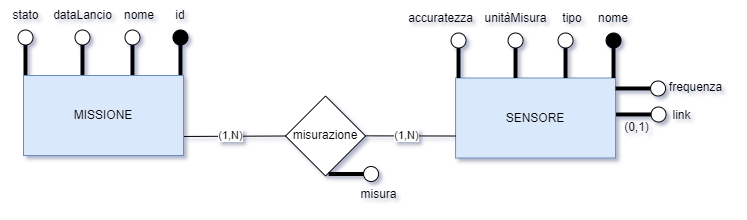


Figura 8 Modello E/R iniziale per la gestione della misurazione dei dati relativi alle missioni

### Raffinamenti proposti

Per la gestione delle misurazioni effettuate dai sensori a bordo dei razzi durante una missione si può gestire diversamente andando a reificare la relazione “misurazione”. Inoltre si gestisce il fatto che se il sistema si trovi in stato di “fail”, dunque non venga effettuata alcuna rilevazione da parte della sensoristica, attraverso la scelta (0,N).

### Schema concettuale parziale

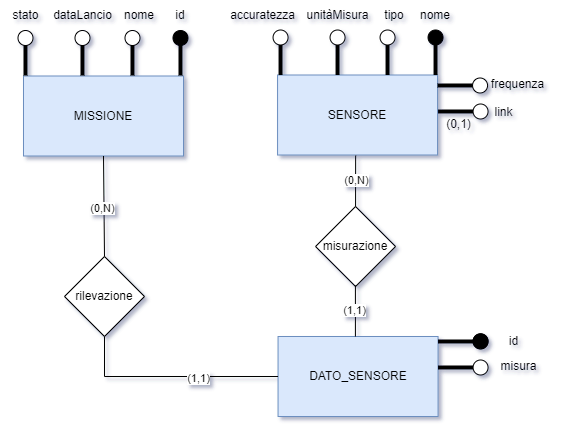
Dopo l’applicazione dei raffinamenti proposti e l’inserimento degli attributi necessari per le varie entità, lo schema E/R parziale risulta essere di questo tipo:

Figura 9 Modello E/R parziale per la gestione della misurazione dei dati delle missioni

## Schema concettuale finale

Figura 10 Modello E/R finale completo

## **Progettazione logica**

### Stima del volume dei dati

Si fornisce in questa fase una tabella contenente il numero medio di istanze di ogni entità e associazione dello schema globale: i valori di carico fanno riferimento alle partecipazioni medie riportate nella figura precedente.

|  |  |  |
| --- | --- | --- |
| Concetto | Tipo (Entity/Relationship) | Volume |
| Obiettivo | E | 50 |
| Luogo | E | 5 |
| Missione\_Obiettivo | R | 30 |
| Payload | E | 5 |
| Missione | E | 15 |
| Dato\_Sensore | E | 200.000 |
| Materiale | E | 10 |
| Razzo | E | 10 |
| Missione\_fornitore | R | 200 |
| Missione\_consulenza | R | 250 |
| Missione\_finanziatore | R | 100 |
| Azienda\_Fornitrice | E | 20 |
| Azienda\_Consulenziale | E | 10 |
| Azienda\_Finanziatrice | E | 5 |
| Sensore | E | 60 |
| Motore | E | 35 |
| Razzo\_Sensore | R | 1.000 |
| Simulazione | E | 200 |
| Paracadute | E | 50 |
| Dipartimento | E | 8 |
| Team | E | 10 |
| Incarico | E | 200 |
| Ruolo | E | 5 |
| Membro | E | 150 |

### Descrizione delle operazioni principali e stima della loro frequenza

Le operazioni da effettuare sono quelle già elencate nella fase di analisi. Segue una tabella riportante la loro descrizione e relativa frequenza:

Siccome sono state richieste l’aggiunta, la modifica e la rimozione di ogni tipo di entità, nella tabella sottostante vengono prese in considerazione solo alcuni esempi, in modo da poterne effettivamente valutare la frequenza correttamente.

|  |  |  |
| --- | --- | --- |
| Codice | Operazione | Frequenza |
| 1 | Aggiunta di un nuovo membro | 20 ogni semestre |
| 2 | Aggiunta/modifica di un nuovo incarico | 30 ogni semestre |
| 3 | Aggiunta di un razzo con relativa missione | 1 ogni semestre |
| 4 | Aggiunta di una nuova azienda finanziatrice (analogo per le altre tipologie di aziende) | 5 all’anno |
| 5 | Salvataggio dei dati dai sensori durante una missione | 200.000 ogni missione |
| 6 | Aggiunta e modifica degli obiettivi relativi ad una missione spaziale | 10 ogni anno |
| 7 | Selezionare gli obiettivi delle relative missioni |  |
| 8 | Selezionare i sensori associati ai vari razzi |  |
| 9 | Selezione dei dati rilevati dai sensori associati ad una missione raggruppati per sensore |  |
| 10 | Seleziona le missioni associate alle aziende fornitrici (analogo per le aziende di consulenza e finanziatrici) |  |

### Schemi di navigazione e tabelle degli accessi

Sono riportate in seguito le tabelle degli accessi delle operazioni sopra riportate; inoltre, sono stati inseriti i relativi schemi di navigazione, ove ha senso inserirlo dovuta alla maggiore complessità dell’operazione. Al fine del calcolo dei costi, si considerano di peso doppio gli accessi in scrittura rispetto a quelli in lettura.

## **Raffinamento dello schema**

### Eliminazione delle gerarchie

Per l’eliminazione della gerarchia delle tipologie di aziende che possono offrire benefici, si è deciso di eliminare l'entità madre "azienda" e far coesistere le entità figlie in quanto è necessario mantenere i vari parametri relativi ad ogni entità, come riportato in [figura 5](#_Schema_concettuale_parziale).

### Eliminazione degli attributi compositi

Nello schema sono presenti gli attributi composti "Luogo" e "Obiettivo" si è deciso di trasformarli in entità e associarli attraverso relazioni all'entità "missione". Per quanto riguarda invece "Dimensione" si è deciso di collassare gli attributi figli nelle entità interessate, ovvero "razzo" e "payload", in quanto rappresentano caratteristiche affini all'entità da modellare. Tutto questo si può notare in [figura 2](#_Progettazione_dello_schema) e in [figura 3](#_Schema_concettuale_parziale_1).

### Scelta delle chiavi primarie

Nello modello E/R finale completo sono già evidenziate senza ambiguità tutte le chiavi primarie tutte le entità.

### Eliminazione degli identificatori esterni

Nello schema E/R sono eliminate le seguenti relazioni (in alcuni casi le chiavi esterne per evitare problemi di ambiguità verranno rinominate del tipo “**id\_<NOME\_ENTITA’>**”):

* “locazione”, tra “LUOGO” e “MISSIONE”, importando “idLuogo” in “MISSIONE”;
* “spedizione”, tra “PAYLOAD” e “MISSIONE”, importando “idPayload” in “MISSIONE”;
* “rilevazione”, tra “DATO\_SENSORE” e “MISSIONE”, importando “idMissione” in “DATO\_SENSORE”;
* “misurazione”, tra “DATO\_SENSORE” e “SENSORE”, importando “idSensore” in “DATO\_SENSORE”;
* “lancio”, tra “MISSIONE” e “RAZZO”, importando “nomeRazzo” in “MISSIONE”;
* “costruzione”, tra “RAZZO” e “MATERIALE”, importando “nomeMateriale” in “RAZZO”;
* “montaggio”, tra “RAZZO” e “MOTORE”, importando “nomeMotore” in “RAZZO”;
* “esecuzione”, tra “SIMULAZIONE” e “RAZZO”, importando “nomeRazzo” in “SIMULAZIONE”;
* “salvataggio”, tra “PARACADUTE” e “RAZZO”, importando “idParacadute” in “RAZZO”;
* “composizione”, tra “DIPARTIMENTO” e “TEAM”, importando “idDipartimento” in “TEAM”;
* “creazione”, tra “TEAM” e “SIMULAZIONE”, importando “nomeTeam” in “SIMULAZIONE”;
* “appartenenza”, tra “TEAM” e “INCARICO”, importando “nomeTeam” in “INCARICO”;
* “assegnazione”, tra “MEMBRO” e “INCARICO”, importando “idMembro” in “INCARICO”;
* “detiene”, tra “RUOLO” e “INCARICO”, importando “nomeRuolo” in “INCARICO”;

## Analisi delle ridondanze

In fase di analisi e progettuale non sono stati introdotti elementi di ridondanza in quanto si è ritenuta la struttura attuale abbastanza organizzata e leggera, sicuramente con qualche possibile ottimizzazione in futuro per la gestione di grandi quantità di dati provenienti dalla sensoristica montata a bordo dei razzi.

### Traduzione di entità e associazioni in relazioni

missioni(id, nome, note, dataLancio, stato, idLuogo: Luogo, idPayload: Payload, idRazzo: Razzo)

razzi(nome, massa, lunghezza, larghezza, altezza, idMotore: Motore, idParacadute: Paracadute, idMateriale: Materiale)

motori(nome, produttore, spinta, impulso, durataCombustione, massa, link\*)

sensori(nome, tipo, unitaMisura, accuratezza, frequenza, link\*)

datiSensori(id, valore, timestamp, idSensore: Sensore, idMissione: Missione)

razziSensori(idRazzo: Razzo, idSensore: Sensore)

paracaduti(nome, modello, diametro)

materiali(nome, note)

payload(id, nome, massa, lunghezza, larghezza, altezza)

simulazioni(id, nome, link\*, idRazzo: Razzo, idTeam: Team)

luoghi(id, stato, regione, città, via, civico)

obiettivi(id, nome)

missioniObiettivi(idMissione: Missione, idObiettivo: Obiettivo)

dipartimenti(nome)

team(nome, idDipartimento: Dipartimento)

membri(id, nome, cognome, email, password)

ruoli(nome)

incarichi(dataInizio, dataFine\*, idMembro: Membro, idRuolo: Ruolo, idTeam: Team)

aziendeFornitrici(nome, note, quantità, link\*)

missioniFornitori (idMissione: Missione, idAzienda: aziendeFornitrici)

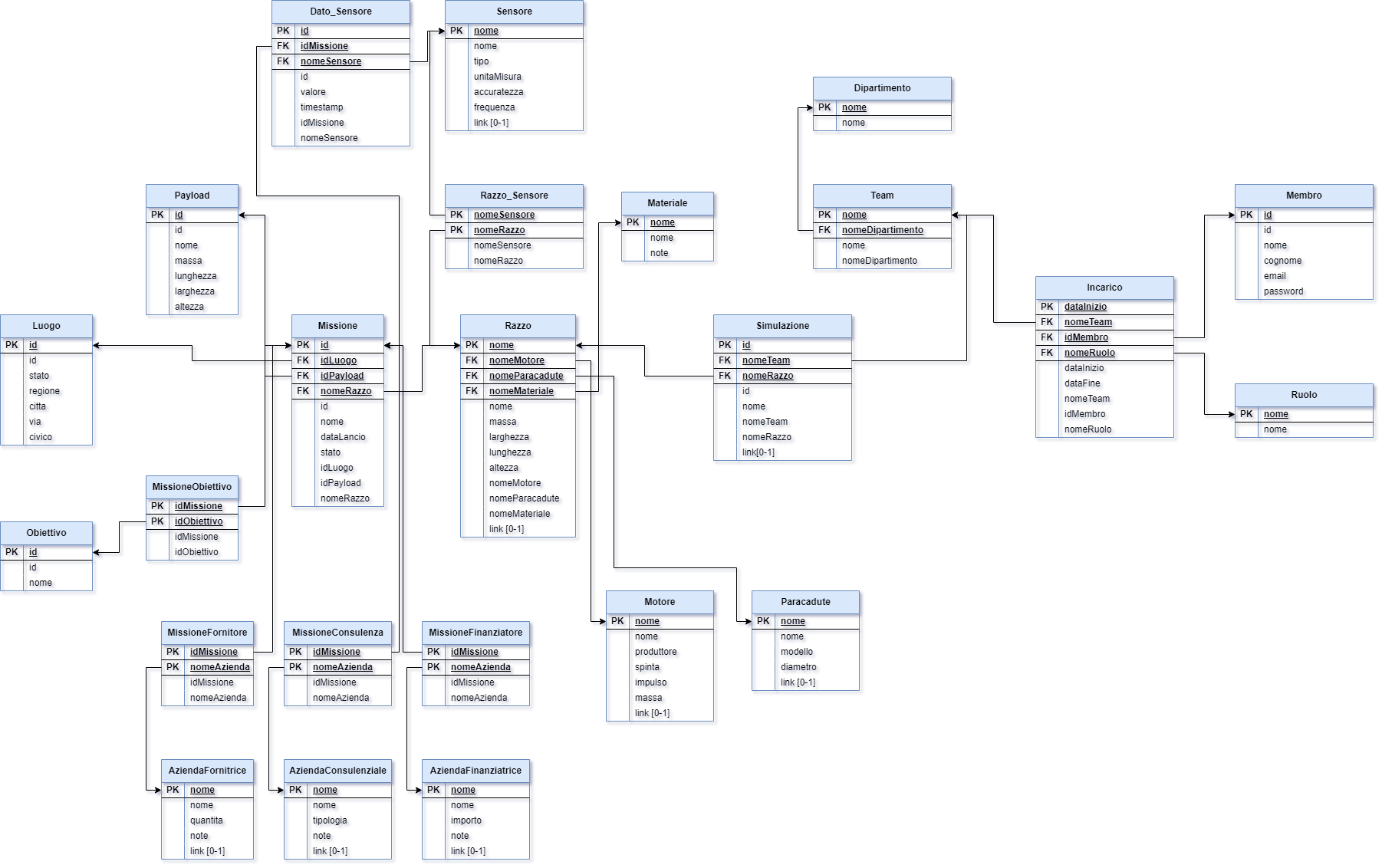
aziendeConsulenziali(nome, note, tipologia, link\*)

missioniConsulenze(idMissione: Missione, idAzienda: aziendeConsulenziali)

aziendeFinanziatrici(nome, note, import, link\*)

missioniFinanziatori(idMissione: Missione, idAzienda: aziendeFinanziatrici)

## Schema relazionale finale



## Traduzione delle operazioni in query SQL

#### Inserimento di tutte le entità riportate nello schema logico

Qui di seguito ne vengono riportati solo alcuni esempi in quanto analogo per le altre entità dello schema logico:

-- Inserimento di un nuovo Membro

INSERT INTO membro (id, nome, cognome, email, password)

VALUES (?, ?, ?, ?, ?);

-- Inserimento di un nuovo Ruolo

INSERT INTO ruolo (nome)

VALUES (?);

#### Aggiornamento di tutte le entità riportate

Qui di seguito ne vengono riportati solo alcuni esempi in quanto analogo per le altre entità dello schema logico:

-- Aggiornamento di un membro

UPDATE membro

SET nome = ?, cognome = ?, email = ?, telefono = ?, password = ?

WHERE id = ?;

#### Eliminazione di tutte le entità riportate

-- Eliminazione di un membro

DELETE FROM membro

WHERE id = ?;

-- Eliminazione di un team

DELETE FROM team

WHERE nome = ?;

#### Selezionare gli obiettivi delle relative missioni

SELECT missione.nome AS missione, obiettivo.nome AS obiettivo

FROM missione

JOIN missione\_obiettivo ON missione.id = missione\_obiettivo.idMissione

JOIN obiettivo ON missione\_obiettivo.idObiettivo = obiettivo.id;

#### Selezionare i sensori associati ai vari razzi

SELECT razzo.nome AS razzo, sensore.nome AS sensore

FROM razzo

JOIN razzo\_sensore ON razzo.nome = razzo\_sensore.nomeRazzo

JOIN sensore ON razzo\_sensore.nomeSensore = sensore.nome;

#### Selezione dei dati rilevati dai sensori associati ad una missione raggruppati per sensore

SELECT missione.nome AS missione, dato\_sensore.nomeSensore AS sensore, dato\_sensore.valore, dato\_sensore.timestamp

FROM dato\_sensore

JOIN missione ON missione.id=dato\_sensore.idMissione

WHERE missione.id = 1

ORDER BY dato\_sensore.nomeSensore, dato\_sensore.timestamp ASC;

#### Seleziona le missioni associate alle aziende fornitrici (analogo per le aziende di consulenza e finanziatrici)

SELECT missione.nome AS missione,

FROM missione

JOIN missione\_fornitore ON missione.id = missione\_fornitore.idMissione

JOIN azienda\_fornitrice ON missione\_fornitore.nomeAzienda = azienda\_fornitrice.id

WHERE azienda\_fornitrice.id = (?);

## Progettazione dell’applicazione

### Descrizione dell’architettura dell’applicazione realizzata

I requisiti dell’intervista iniziale vengono soddisfatti grazie all’implementazione di una web app ad hoc per il team in grado di poter monitorare facilmente lo storico delle missioni e delle sue caratteristiche.

La web app è stata realizzata in **Python** con il framework **Flask** per la parte web e **SQLAlchemy (ORM)** per la parte di interazione con il database.

La scelta del linguaggio è dovuta dalla necessità di poter avere a disposizione uno strumento in grado di manipolare grandi quantità di dati come nel caso dell’ottenimento e salvataggio dei dati derivanti dai sensori.

Il database risiede in locale e come **DBMS** è stato utilizzato **MySQL**.

La correttezza dei dati inseriti viene verificata dal DBMS sfruttando i check, in modo da cercare di ridurre al minimo i controlli che l’applicazione deve svolgere.

Viene implementata anche l’autenticazione degli utenti attraverso login, sicurezza degli endpoint attraverso accesso non autorizzato, gestione delle tipologie di risorse visualizzabili dipendentemente dal ruolo dell’utente, hashing per la password degli utenti inserita in fase di registrazione.

La fase di registrazione degli utenti è eseguita da un responsabile e solo successivamente a questa il nuovo utente potrà accedere alla web app.

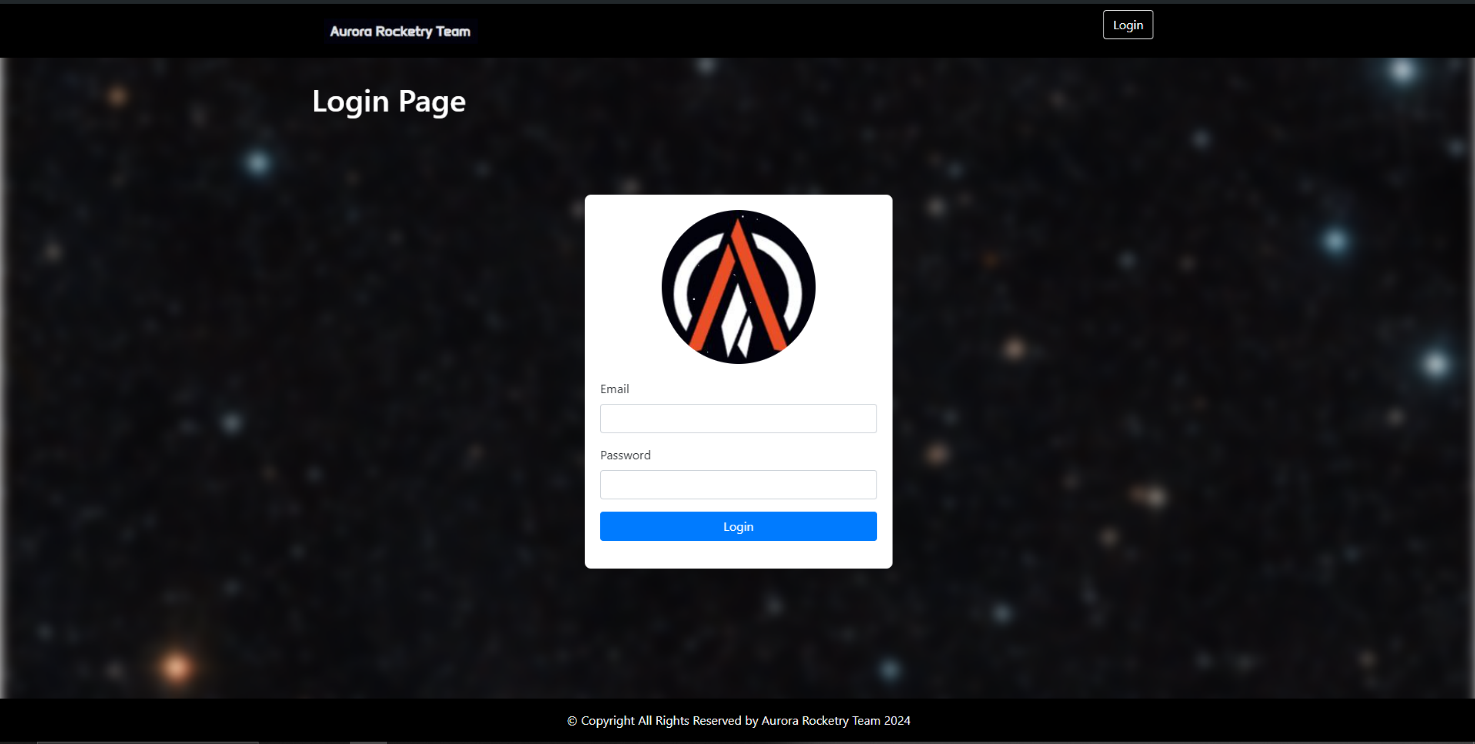


Figura 11 Pagina di Login iniziale

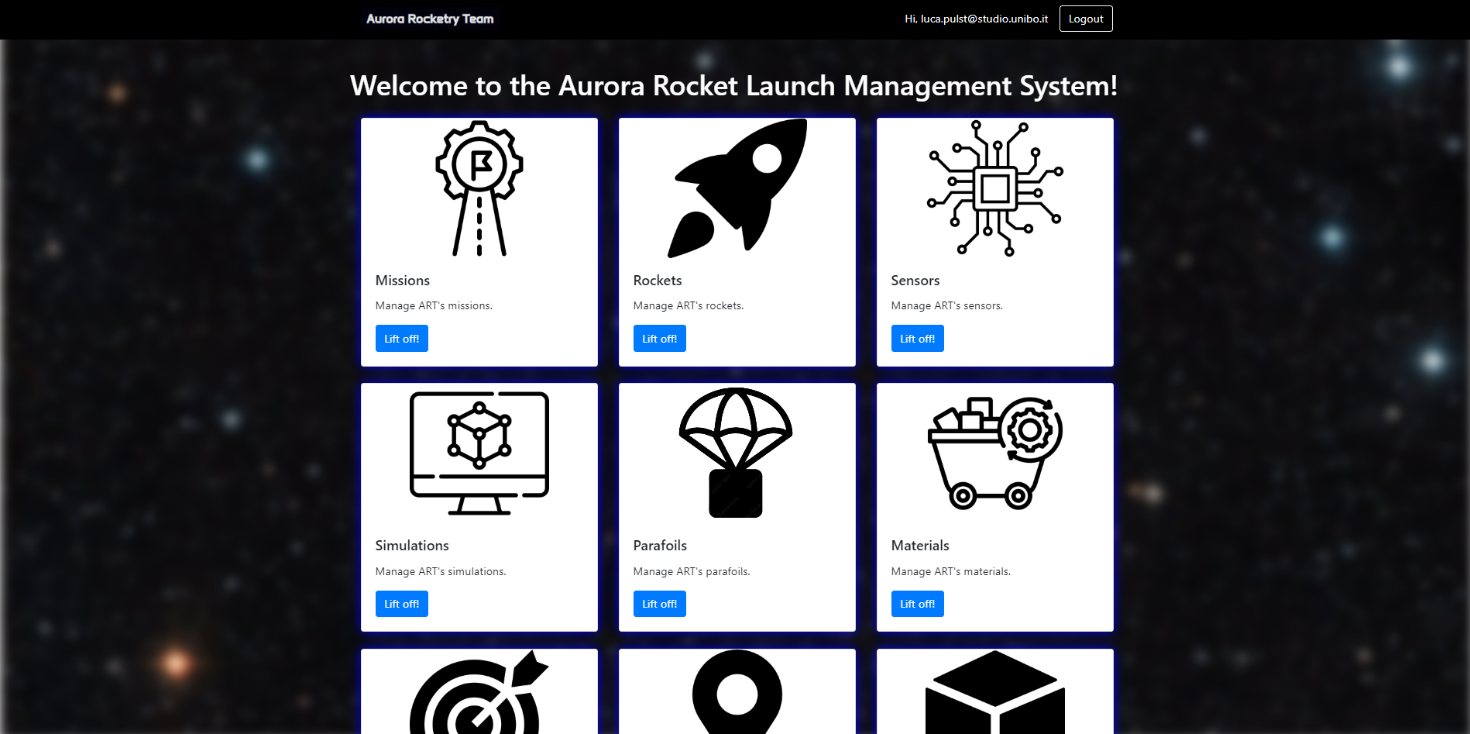
Si riporta qui di seguito solo un esempio della gestione di tutte le entità mostrate in fase progettuale.

Figura 12 Homepage

## 

Figura 13 Gestione delle informazioni relative alle missioni

## 

Figura 14 Gestione dei razzi con associazione dei relativi sensori

## 

Figura 15 Visualizzazione dei dati categorizzati per sensore di una missione

## Link

Link al sito ufficiale di Aurora Rocketry Team Unibo:

<https://www.aurorarocketry.eu/>