

# **Piattaforma di Sport- Analytics**

**ANNO SCOLASTICO 2024/2025**

**MATERIA: TPSI**

**Sparamonti Nicolò, Scucchia Gabriele, Gilardi Marco**



# INDICE:

## 1. Introduzione e Descrizione del Problema

- 1.1. Contesto
- 1.2. Problema Affrontato
- 1.3. Obiettivi del Progetto
- 1.4. Suddivisione dei compiti e idee
- 1.5. Svolgimento del progetto

## 2. Funzionalità Implementate

- 2.1. Frontend
- 2.2. Data Engineering
- 2.3. Backend

## 3. Mockup e Progettazione UI

- 3.1. Design System
- 3.2. Mappa dell'Applicazione
  - \* HomePage ( / )
  - \* DriversPage ( /drivers )
  - \* DriverDetailsPage ( /drivers/:id )
  - \* CalendarPage ( /calendar )
- 3.3. User Experience Design

## 4. Algoritmo di Analisi dei Dati Sportivi

- 4.1. Descrizione dell'Algoritmo
- 4.2. Implementazione
- 4.3. Valutazione delle Performance

## **5. Procedura di Data Engineering**

5.1. Fonti dei Dati

5.2. Metodologia di Utilizzo API

5.3. Processo di Data Cleaning

5.4. Struttura del Database

## **6. Endpoint del Web Service**

6.1. API Endpoints

6.2. Sicurezza e Gestione Errori

## **7. Tecnologie Utilizzate**

7.1. Frontend

7.2. Backend

7.3. Data Engineering

7.4. Strumenti di Sviluppo

## **8. Conclusioni**

8.1. Risultati Ottenuti

8.2. Limitazioni Attuali

## **9. Appendici**

9.1. Riferimenti

9.2. Glossario

9.3. Link GitHub

# Relazione Tecnica: Piattaforma di Sport-Analytics

## Titolo: Piattaforma di Sport-Analytics: Sistema integrato per l'analisi delle prestazioni sportive IndyCar

**Autori:** Marco Gilardi, Gabriele Scucchia, Nicolò Sparamonti

---

### 1. Introduzione e Descrizione del Problema

#### 1.1 Contesto

Nel mondo delle competizioni automobilistiche, l'analisi dei dati è diventata un elemento fondamentale per comprendere le prestazioni dei piloti e l'andamento delle stagioni. Le IndyCar, in particolare, generano una grande quantità di dati che necessitano di strumenti specializzati per essere analizzati e visualizzati in modo efficace. La crescente importanza degli strumenti di sport analytics nel motorsport richiede soluzioni moderne e accessibili per appassionati, analisti e professionisti del settore.

#### 1.2 Problema Affrontato

Rendere disponibile e facilmente accessibile un sistema di visualizzazione dei dati sportivi e calcolo di statistiche riguardanti le IndyCar. Attualmente, i dati sono spesso dispersi e non facilmente consultabili in un formato integrato e interattivo.

#### 1.3 Obiettivi del Progetto

Creare un sito web per l'analisi e la visualizzazione dei dati delle IndyCar. Il sistema deve permettere di:

- Visualizzare dati sulle gare e sui risultati
- Monitorare le prestazioni dei piloti
- Ottenere approfondimenti analitici attraverso dashboard interattive
- Fornire statistiche avanzate e previsioni

#### 1.4 Suddivisione dei compiti e idee:

- Front-End: Gabriele Scucchia / Nicolò Sparamonti
- Back-End: Gabriele Scucchia / Nicolò Sparamonti

- Sparamonti:
  - Homepage 2025
  - Filtri classifica piloti per stagione
  - Ordinamento campi classifica piloti
- Scucchia:
  - Filtri e stato gare
  - Pagina dettagli piloti
- Data Engineering: Marco Gilardi
- Relazione tecnica: Nicolò Sparamonti

## 1.5 Svolgimento del progetto:

Lo svolgimento del progetto ha visto una chiara suddivisione dei ruoli.

Il lavoro sul Front-End e sul Back-End è stato caratterizzato da una stretta e costante collaborazione. Le fasi di sviluppo sono state condotte quasi sempre in parallelo, con un continuo scambio di idee, suggerimenti e correzioni reciproche. Questa sinergia ha permesso di integrare efficacemente le diverse componenti e di affrontare in modo congiunto le sfide emerse durante la programmazione, garantendo coerenza e funzionalità tra le diverse sezioni del sito.

Marco Gilardi si è occupato in modo autonomo ma integrato del Data Engineering, gestendo la raccolta, pulizia e strutturazione dei dati per il database.

Infine Nicolò Sparamonti ha curato anche la redazione della relazione tecnica.

---

## 2. Funzionalità Implementate

### 2.1 Frontend

Dashboard interattiva per visualizzazione dati sportivi:

- Homepage con statistiche principali della stagione corrente
- Vista leader della classifica con pilota in testa
- Top 10 piloti con statistiche dettagliate
- Pilota favorito con probabilità di vittoria calcolata

Interfaccia per il monitoraggio delle prestazioni dei piloti:

- Pagina dedicata ai piloti con classifica ordinabile

- Filtri per stagione (2017-2025)
- Dettagli completi di ogni pilota con storico gare

Strumenti di visualizzazione per approfondimenti analitici:

- Calendario interattivo delle gare con filtri avanzati
- Stati delle gare in tempo reale

## 2.2 Data Engineering

Sistemi di raccolta dati automatizzata:

- **Script Python** per l'estrazione dati da API Sportradar
- Integrazione con MongoDB per salvataggio dati

Processi di pulizia e normalizzazione dei dati (in Python):

- Rimozione virgole dai nomi piloti
- Conversione posizioni in formato numerico
- Gestione stati speciali (DNF, gare in corso)
- Validazione e formattazione date

Sistema di archiviazione su database non-relazionale:

- Database MongoDB "IndyCar"
- Struttura ottimizzata per queries complesse
- Pipeline di aggregazione per statistiche

## 2.3 Backend

Sistema di analisi statistica per piloti e gare:

- Calcolo punti totali, vittorie, podi per pilota
- Statistiche per stagione e generali
- Classifiche dinamiche

Algoritmo di calcolo probabilità vittoria:

- Formula pesata basata su:
  - Win rate storico (40%)
  - Podium rate storico (30%)
  - Punti stagione corrente (20%)
  - Vittorie stagione corrente (10%)

API RESTful per comunicazione frontend:

- Endpoint per dati homepage
- Endpoint per calendario gare
- Endpoint per classifica piloti
- Endpoint per dettagli piloti
- **I dati vengono recuperati dal database MongoDB**

Gestione errori e logging completo

---

### 3. Mockup e Progettazione UI

#### 3.1 Design System



- Colori dinamici: Gradiente racing con effetti "racing stripe"
- Animazioni fluide: Hover effects, micro-animazioni
- Layout moderno: Card-based, responsive
- Tipografia racing: Font ispirati al motorsport

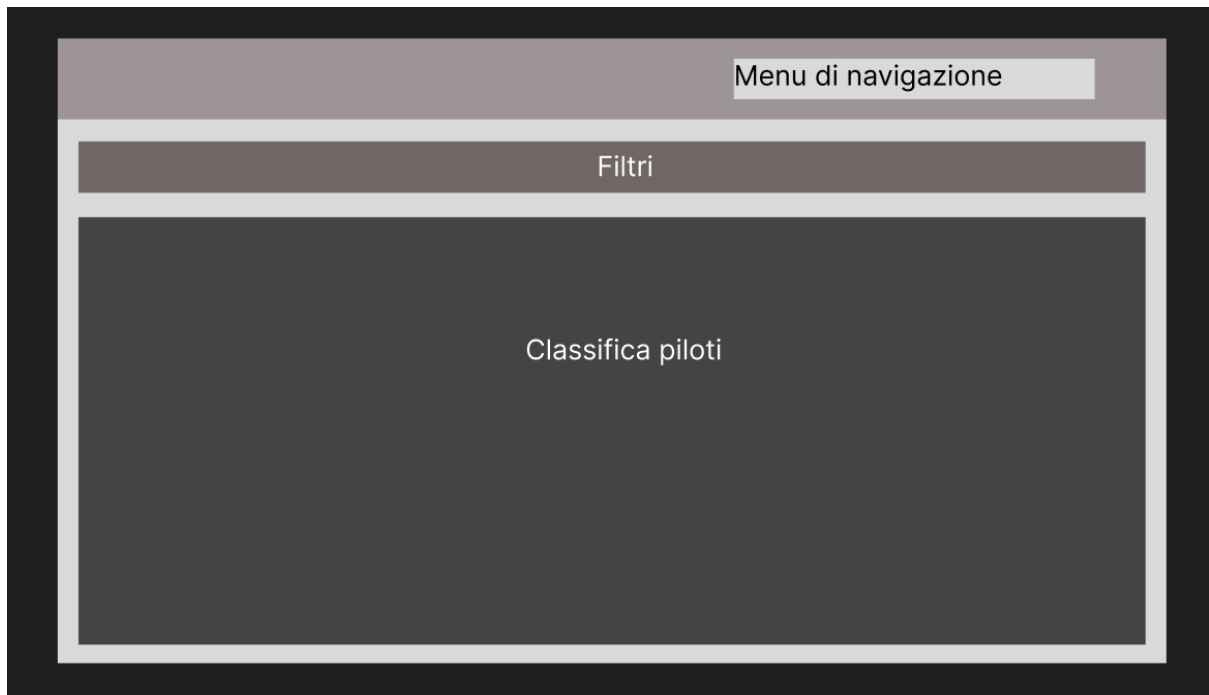
### 3.2 Mappa dell'Applicazione

#### HomePage (/)

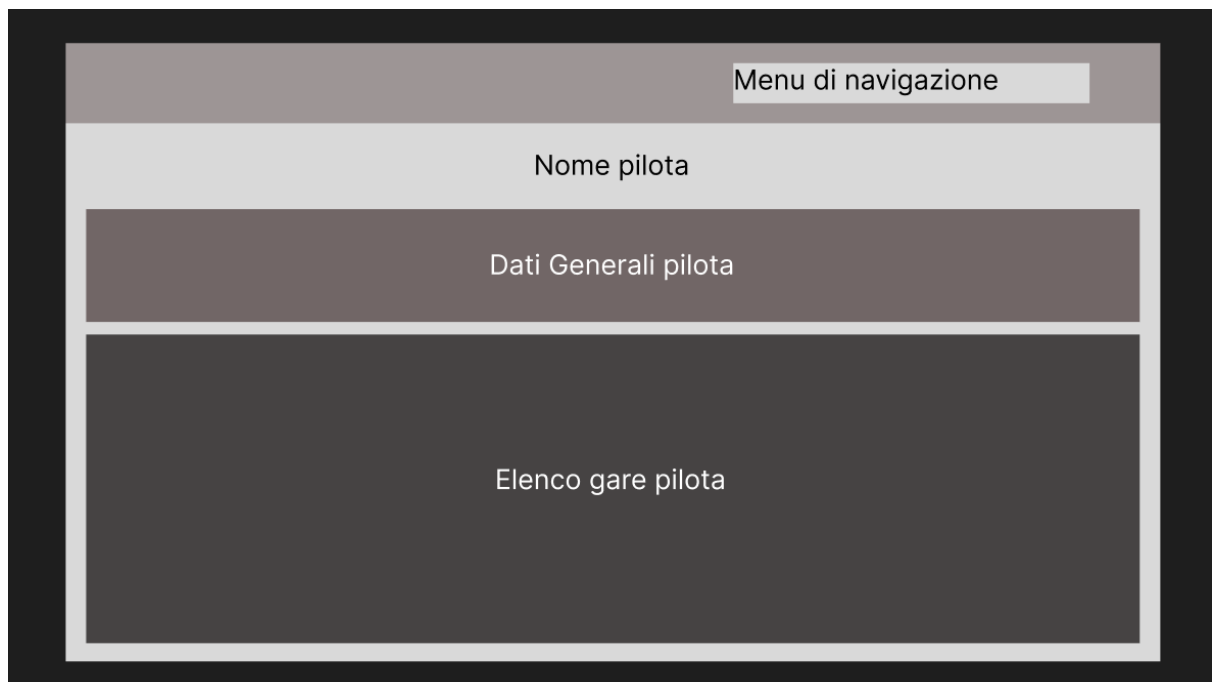


- Leader Dashboard
- Top 10 Piloti
- Prossima Gara
- Pilota Favorito

#### DriversPage (/drivers)



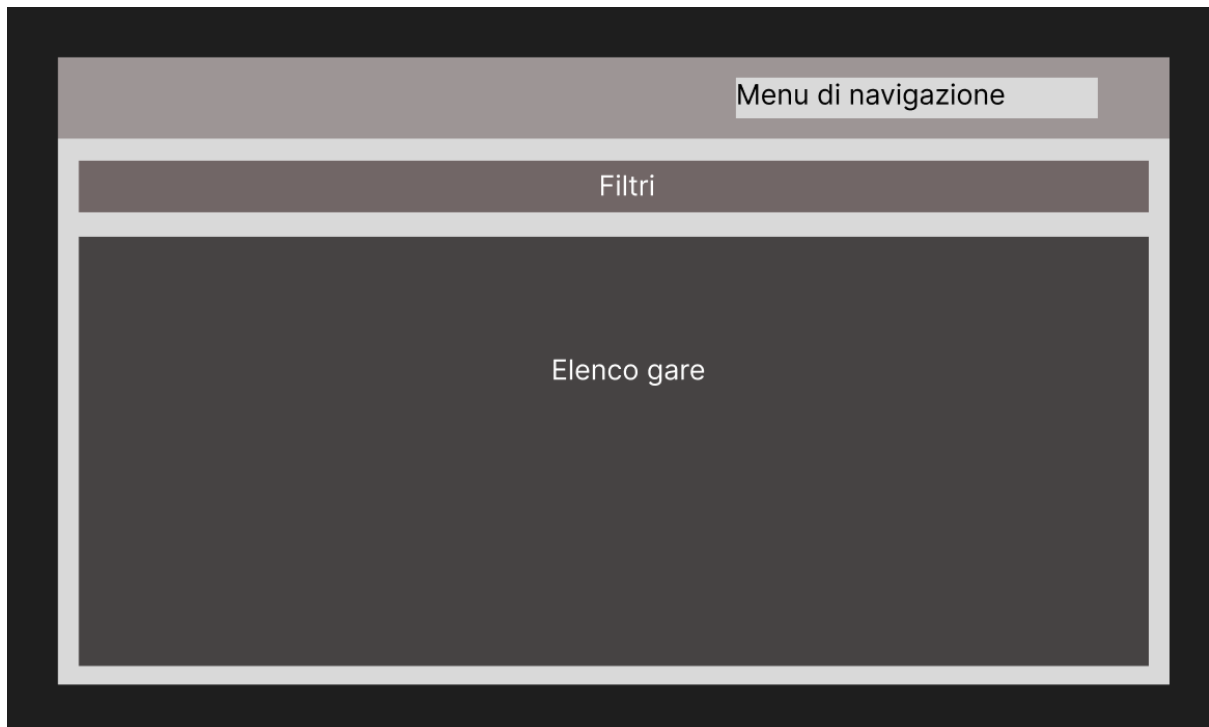
- Selettore Stagione
- Tabella Classifica
- → DriverDetailsPage (</drivers/:id>)



- Statistiche Generali

- Storico Gare

## CalendarPage (/calendar)



- Filtri Avanzati
- Griglia Gare
- Stati Real-time

### 3.3 User Experience Design

- Navigazione intuitiva
  - Feedback visivo
  - Accessibilità e performance
-

## 4. Algoritmo di Analisi dei Dati Sportivi

### 4.1 Descrizione dell'Algoritmo

**Probabilità Vittoria** =  $(WinRate \times 0.4) + (PodiumRate \times 0.3) + (PuntiAttuali \times 0.2) + (VittorieAttuali \times 0.1)$

### 4.2 Implementazione

Nel **backend Node.js**, con aggregazione MongoDB:

- Raccoglie dati storici
- Calcola metriche base
- Applica pesi e normalizza il risultato

### 4.3 Valutazione delle Performance

- Accuratezza su dati dal 2017
  - Precisione con 4 parametri bilanciati
  - Aggiornamento dinamico
- 

## 5. Procedura di Data Engineering

### 5.1 Fonti dei Dati

API Sportradar – Sezione IndyCar

### 5.2 Metodologia di Utilizzo API

Script Python:

- Librerie: `requests`, `pymongo`
- Acquisizione automatica dati
- Retry logic e gestione rate limiting
- Inserimento su MongoDB

### 5.3 Processo di Data Cleaning

In Python:

- Normalizzazione nomi
- Validazione posizioni e gestione DNF
- Formato ISO 8601
- Gestione null e deduplicazione

### 5.4 Struttura del Database

Database: **MongoDB "IndyCar"**

#### Collection Season

```
{
  "_id": ObjectId,
  "season_year": "2025",
  "description": "IndyCar Series 2025",
  "races": [race_ids...]
}
```

#### Collection Race

```
{
  "_id": ObjectId,
  "race_id": "sr:race:123456",
  "description": "Indianapolis 500",
  "scheduled": "2025-05-25T17:00:00Z",
  "status": "Scheduled",
  "venue": {
    "name": "Indianapolis Motor Speedway",
    "city": "Indianapolis",
    "country": "USA"
  },
  "season_year": "2025",
  "drivers": [driver_results...]
}
```

#### Collection Drivers

```
{
  "_id": ObjectId,
```

```
"driver_id": "sr:driver:123456",
"nome": "Scott Dixon",
"numero_auto": 9,
"statistics": {
  "punti_totali": 485,
  "vittorie": 12,
  "podii": 28,
  "gare_disputate": 45
}
}
```

---

## 6. Endpoint del Web Service

### 6.1 API Endpoints

Endpoint	Metodo	Descrizione	Parametri	Risposta
<code>/calendar</code>	GET	Lista completa gare con stati	Nessuno	JSON di gare
<code>/drivers</code>	GET	Classifica piloti	<code>season</code> (opzionale)	JSON piloti
<code>/drivers/:driverId</code>	GET	Dettagli pilota	<code>driverId</code> (path)	Oggetto pilota
<code>/drivers/homepage/2025</code>	GET	Dati dashboard homepage	Nessuno	Leader, top10, favorito

Esempio risposta `/calendar`:

```
[
  {
    "_id": "507f1f77bcf86cd799439011",
    "race_id": "sr:race:123456",
    "description": "Indianapolis 500",
    "scheduled": "2025-05-25T17:00:00Z",
    "status": "Scheduled",
    "venue": {
      "name": "Indianapolis Motor Speedway",
      "city": "Indianapolis",
      "country": "USA"
    }
  }
]
```

]

## 6.2 Sicurezza e Gestione Errori

- Validazione input e sanitizzazione query
  - Try-catch su tutti gli endpoint
  - Logging avanzato
  - Codici HTTP coerenti
- 

## 7. Tecnologie Utilizzate

### 7.1 Frontend

- **React + TypeScript**
- **React Router**
- **Styled-components**, CSS
- Componenti personalizzati per grafici e griglie

### 7.2 Backend

- **Node.js**, **Express.js**
- **MongoDB** con driver nativo o Mongoose
- **API RESTful**
- Librerie: **CORS**, **Dotenv**

### 7.3 Data Engineering

- **Python** per data ingestion e cleaning
- Librerie: **requests**, **pymongo**, **schedule**

## 7.4 Strumenti di Sviluppo

- GitHub, VS Code, Chat GPT
- 

## 8. Conclusioni

### 8.1 Risultati Ottenuti

- Sistema completo e funzionante
- UI moderna e responsive
- API performanti
- Database con 9 anni di dati
- Algoritmo predittivo operativo

### 8.2 Limitazioni Attuali

- Dipendenza da Sportradar per aggiornamenti
  - Algoritmo basato solo su dati storici/statistici
- 

## 9. Appendici

### 9.1 Riferimenti

- Sportradar API
- React.dev
- MongoDB Docs
- Express.js Guide



## 9.2 Glossario

- **DNF:** Did Not Finish
- **IndyCar:** Serie americana di monoposto
- **Sportradar:** Provider dati sportivi
- **TypeScript:** Superset di JavaScript
- **UX/UI:** User Experience / Interface

## 9.3 Link GitHub

**Repository:** <https://github.com/Garlics07/Progetto-Gilardi-Scucchia-Sparamonti>