ISTITUTO STATALE DI ISTRUZIONE SUPERIORE
GOBETTI VOLTA

Elaborato Esame di Stato

a.s. 2020/2021

Materie caratterizzanti:

Informatica, Sistemi e Reti

"Autostrade Toscana Manutenzione Infrastrutture"

Viadotto Italia

Docente Referente:

Giuseppe Scaranello

Candidato:

Lorenzo Bartolini

Indice

1. Abstract

Il progetto tratta la realizzazione di un sistema informativo finalizzato alla gestione di sensori per la manutenzione di ponti e viadotti.

Viene, inoltre, previsto lo sviluppo di un portale web che permetta alle società di manutenzione di accedere agli appalti aperti nella regione con la successiva possibilità di eseguirne la manutenzione.

Sono monitorati tre diversi parametri di interesse: l’elettricità, la struttura e l’asfalto.

Per il monitoraggio si utilizza varie tipologie di sensori che comunicheranno il loro stato ad un dispositivo.

Il suddetto analizzerà i valori ricevuti e invierà alla base di dati centralizzata, tramite un canale trasmissivo sicuro, i valori aggregati dei diversi sensori presenti sul posto.

E’ predisposto per il Ministero dei Trasporti un accesso sicuro e diretto ai dati prodotti dai sensori.

I valori dei sensori memorizzati all’interno della base di dati sono espressi con un numero che ne indica la bontà; questo numero varia da 0, valore più basso che indica lo stato peggiore, a 100, valore più alto indicante l’assenza di problemi.

I sensori campionano e comunicano i dati una volta al giorno regolarmente.

2. Architettura Software

Durante la fase di analisi di un progetto viene definita l'architettura software utilizzata che, in questo caso, si basa sul concetto di sistema distribuito.

Per questo progetto è stata sviluppata un'architettura *n-tier* o *multistrato*.

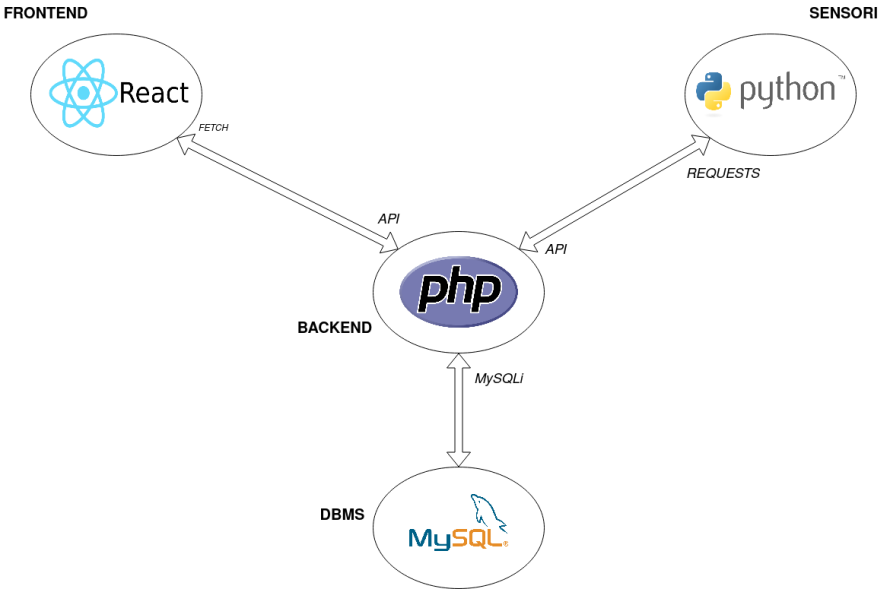

figura 1

Come è possibile vedere nell'immagine sopra, si identificano 4 diverse sezioni divise su 3 strati:

- Presentazione, o frontend
- Logica, o backend
- Risorse

La parte di Frontend è gestita tramite il framework React.

La parte di Backend è affidata al PHP.

Il DBMS è MySQL.

Infine la simulazione dei sensori è eseguita in Python e fa parte dello strato di Risorse.

2.1 Frontend con React.js

Nello specifico, analizzando lo strato di presentazione, solitamente chiamato Frontend, è stato scelto di utilizzare un framework JavaScript: React.js.

Ciò che ha permesso la separazione concettuale tra lo strato di presentazione e quello di logica è proprio l'utilizzo di React.

La differenza tra l'uso di React e lo sviluppo classico di pagine web è che React genere delle SPA, Single Page Application.

Il funzionamento logico di React è descritto nella seguente immagine:

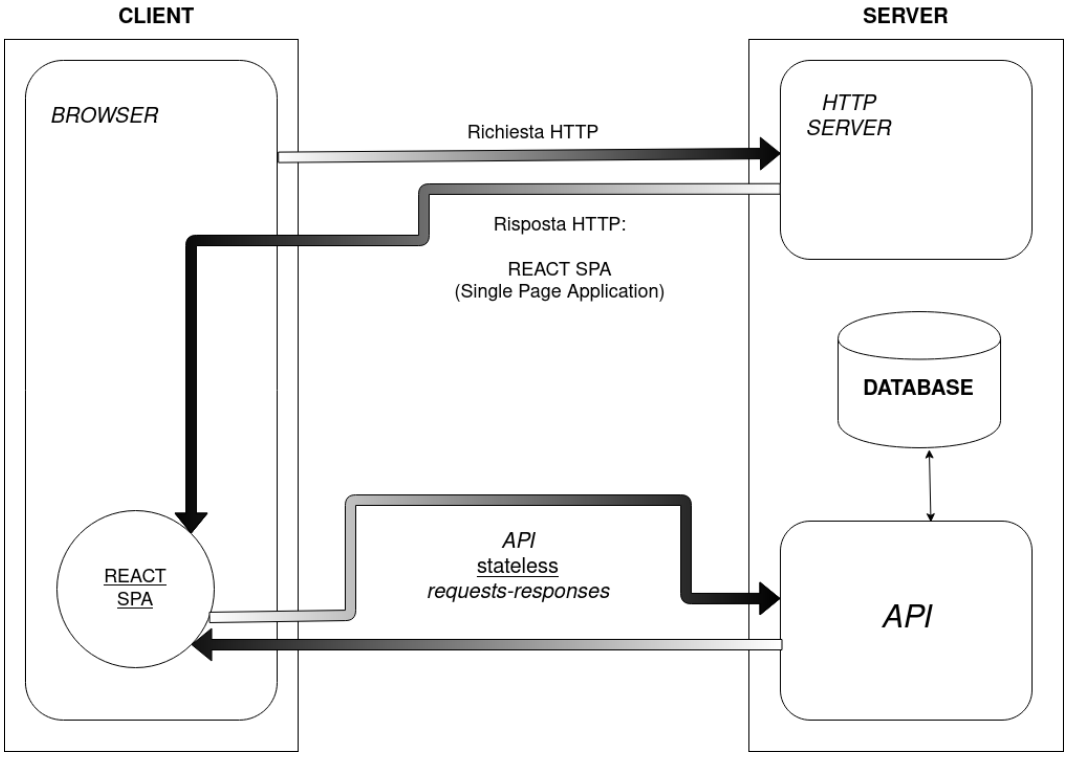

figura 2

Nel momento in cui il browser invia la richiesta al server, questo restituisce al client una pagina web composta da HTML, CSS ed infine il JavaScript, che rende la pagina dinamica tramite l'utilizzo di React.

Una volta che il client riceve la pagina web non avrà più bisogno di comunicare con il Server HTTP; questa è la maggiore differenza con lo sviluppo classico di pagine web.

Nel momento in cui il client ha necessità di alcune specifiche informazioni, queste verranno richieste ad un altro servizio presente sul server: l'API, Application Programming Interface.

La comunicazione con l'API avverrà scambiandosi pacchetti HTTP contenenti dati in formato JSON e non HTML.

Questa scelta progettuale garantisce velocità e fluidità all'intero sito perchè vengono scambiati solo pochi dati ogni volta che è necessario cambiare schermata o visualizzare informazioni diverse.

2.2 Backend con API in PHP

Per quanto riguarda lo strato di logica è stato deciso di codificarlo in linguaggio PHP mediante la realizzazione di API.

Un'API, Application Programming Interface, è un'interfaccia software che permetta lo scambio di dati tra un client ed un server.

Questi due metodi permettono la ricezione di dati dal server, GET, e la possibilità di inviare dati, POST.

Le interfacce realizzate hanno lo scopo di rispondere solo ad una determinata richiesta proveniente da particolari tipi di utenti.

Esistono tre diverse categorie di utente all'interno di questo sistema: ministero, società autostradale, società di manutenzione.

Il processo di accesso all'area riservata è il medesimo per ogni utente, sarà il Frontend a preoccuparsi di mostrare a schermo le informazioni dedicate ad ogni utente.

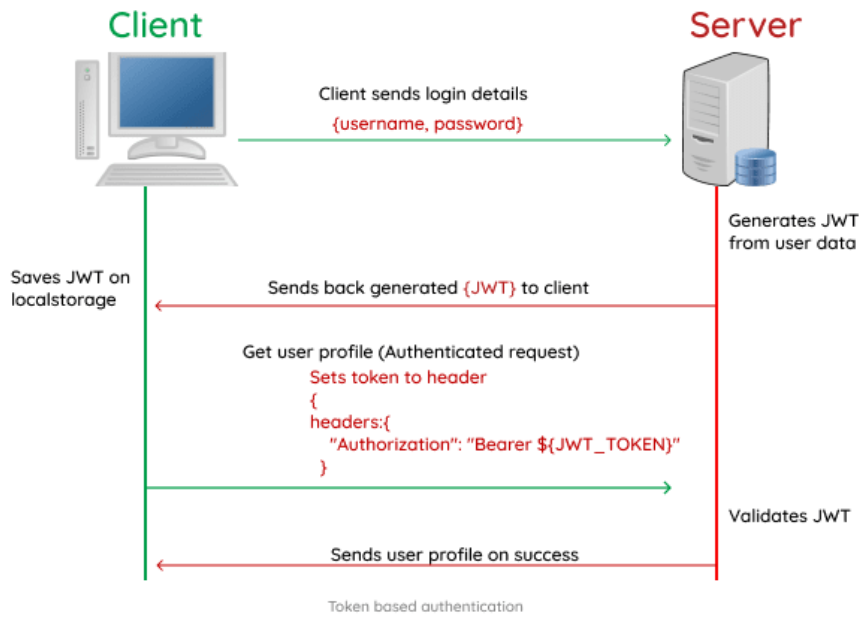

figura 3

Una volta completato l'accesso sarà predisposto un sistema di autenticazione per le richieste successive.

Quello descritto in figura è il processo di autenticazione basato sui token, una stringa alfanumerica generata a partire da una stringa relativa all'utente, per esempio l'email, e da una segreta usata successivamente per la verifica del token stesso.

Questi inoltre hanno una durata per evitare che, una volta generato, non sia più necessario l'accesso tramite credenziali.

Quando il server genera il token viene inviato al client che lo memorizzerà su disco, nel caso abbiano lunga durata permettendo quindi l'accesso al sito senza l'inserimento di credenziali, oppure verrà memorizzato in memoria e durerà per il tempo di navigazione all'interno del sito.

Per proseguire con l'esplorazione del sito, durante le successive richieste all'API, verrà inviato al server il token appena ricevuto.
Il server, per ogni interfaccia, o rotta, che deve essere riservata, andrà a verificare la stringa appena ricevuta tramite quella segreta usata per la sua generazione. Nel caso in cui il token non risulti valido verrà inviata una risposta al client che lo forzerà ad accedere nuovamente tramite credenziali.
Questa scelta progettuale garantisce scalabilità maggiore e la possibilità di accedere da più dispositivi allo stesso utente.

2.3 DBMS e Sensori

La gestione della base di dati è affidata a MySQL.

Lo strato di risorse è composto anche dalla simulazione dei sensori in Python.

Come per React, anche Python, simulando i sensori, invia i dati al server mediante chiamate API al PHP.

Il funzionamento dettagliato di come vengono simulati i sensori in Python sarà effettuato successivamente.

3. Database

3.1 Studio di fattibilità e analisi dei requisiti

Questa fase del ciclo di vita presuppone un'analisi della richiesta.

Si considera che i sensori inviino al server nuovi dati una ed una sola volta al giorno.

Le infrastrutture sono di due tipi:

- Ponti
- Viadotti

Ogni infrastruttura verrà monitorata da almeno un sensore.

Esistono tre tipi di utenti che possono accedere al sito:

- Utente del Ministero dei Trasporti
- Utente della Società Autostradale
- Utente della Società di Manutenzione

3.2 Progettazione Concettuale

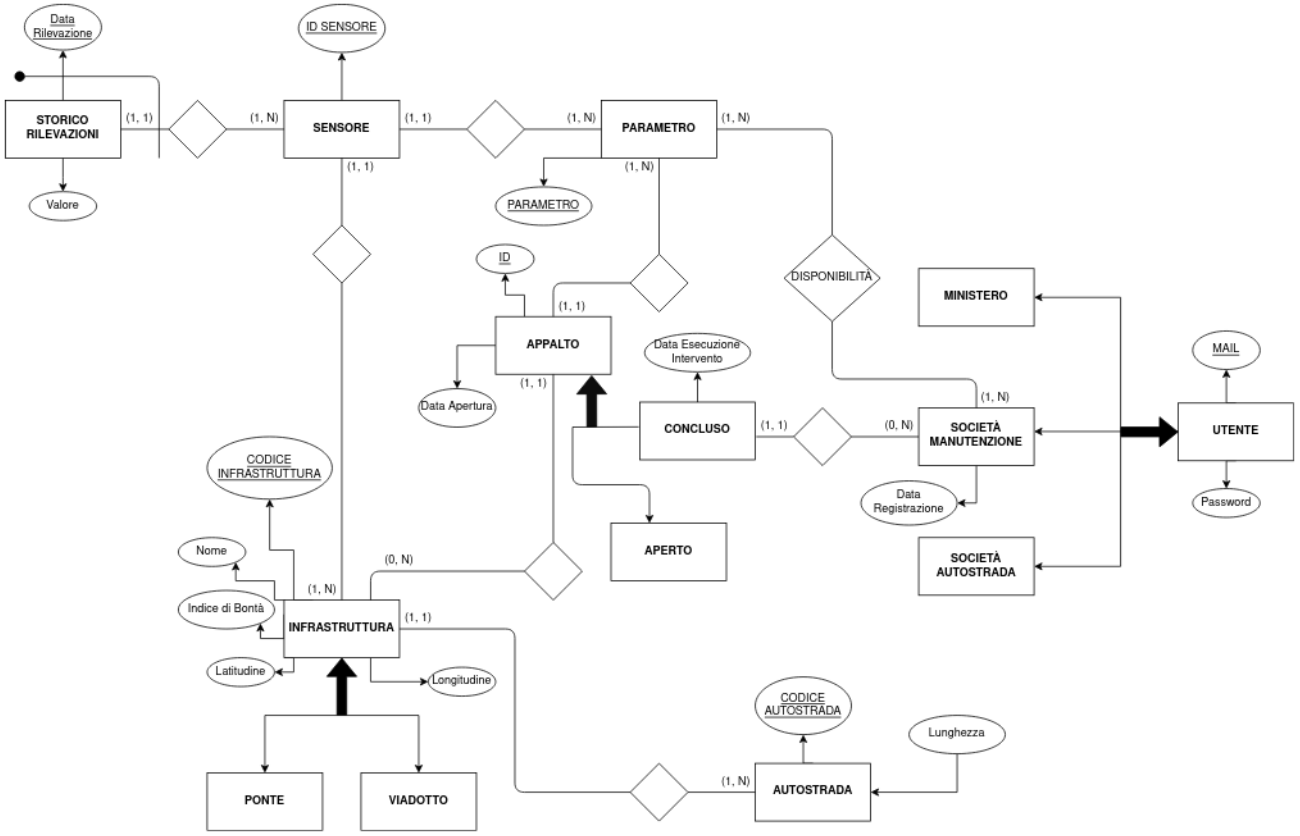

figura 4

Quello rappresentato in figura è il modello E-R completo.

Sono analizzati nel dettaglio le tre entità più importanti:

- Storico Rilevazioni
- Infrastruttura
- Appalto

Storico Rilevazioni è l'entità che racchiude tutti i valori dei sensori.

Per identificare una singola rilevazione si utilizza il codice del sensore stesso, presente come chiave esterna dall'entità Sensore, e la data di rilevazione.

La data è considerata univoca perché i sensori inviano i dati al server una sola volta al giorno.

La valore della rilevazione è memorizzato nel campo Valore.

Infrastruttura è l'entità in cui vengono salvate le informazioni relative alle infrastrutture che sono divise in Ponti e Viadotti. Le infrastrutture sono correlate all'autostrada di appartenenza.

Ogni infrastruttura è identificata univocamente da un Codice numerico progressivo. Sono di interesse anche il Nome e le Coordinate di dove si trova.

Il parametro IndiceBontà è un valore compreso tra 0 e 100 che rappresenta lo stato generale dell'infrastruttura.

Questo valore viene calcolato come la media dell'ultima rilevazione di ogni sensore presente sul ponte o viadotto. Il server aggiorna questo numero ogni volta che un sensore invia un nuovo valore al database.

Appalto è l'entità che gestisce tutti gli appalti del sistema, si dividono in:

- Aperto, appalto che non è stato ancora assegnato ad una società di manutenzione
- Chiuso, identifica l'appalto che ha portato ad un intervento di manutenzione

Dell'appalto è di interesse la data di apertura ed è identificato tramite un id progressivo.

Nello specifico, l'Appalto Chiuso prevede anche una data di esecuzione dell'intervento e l'identificatore della società che l'ha effettuato.

Le società di manutenzione sono correlate all'entità Parametro per tenere traccia della specializzazione di ogni società.

3.3 Progettazione Logica

Si procede ristrutturando lo schema E-R, in figura schema ristrutturato.

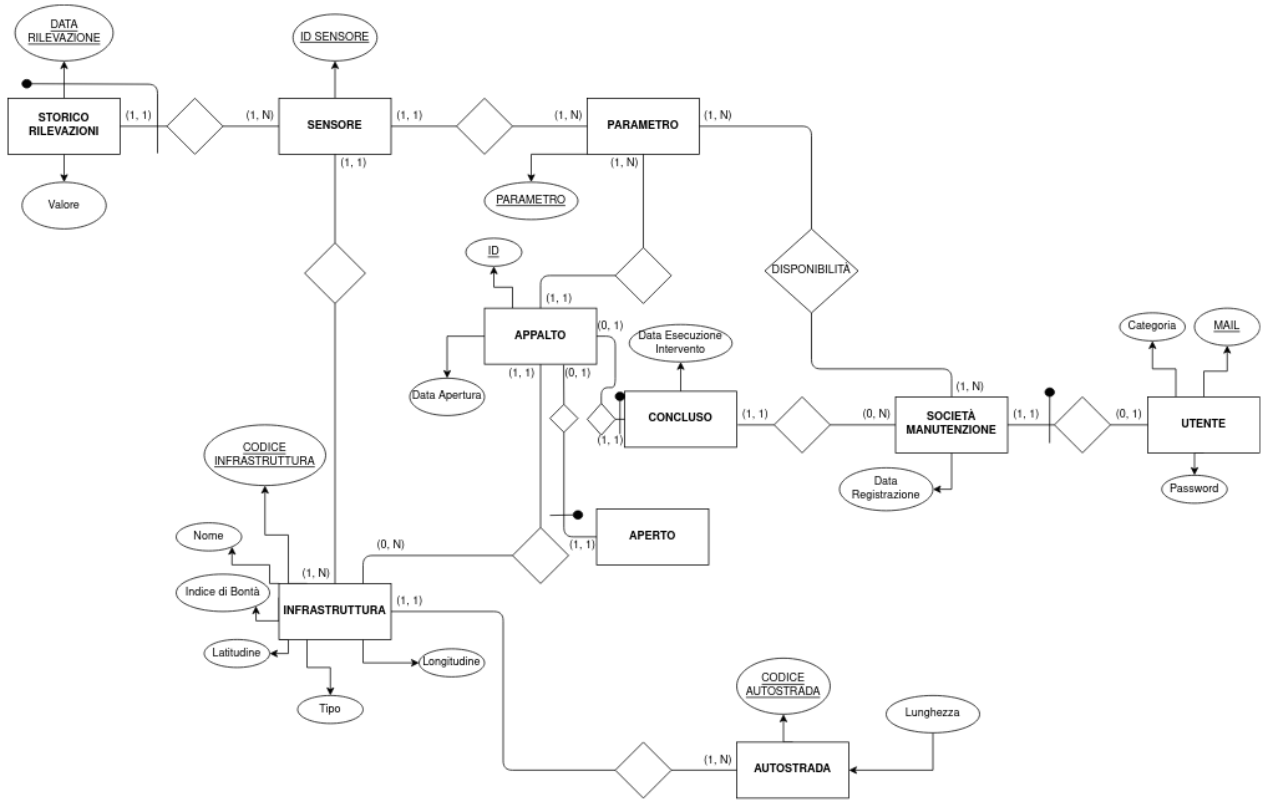

figura 5

La fase di ristrutturazione prevede quattro operazioni:

- Analisi delle ridondanze
    - In seguito ad un'attenta analisi delle prestazioni è stato deciso di mantenere l'attributo Indice di Bontà nell'entità Infrastruttura perchè troppo dispendioso da calcolare ad ogni chiamata
- Accorpamento e separazione di concetti
    - Non necessaria
- Scelta degli identificatori e risoluzione degli attributi multivalore
    - Non necessaria
- Eliminazione delle generalizzazioni

Le tre generalizzazioni presenti sono quelle che riguardano:

- Infrastruttura
- Appalto
- Utente

Quella di *Infrastruttura* è stata risolta eliminando le entità figlie e creando un nuovo campo nel padre che ne identifica il tipo (ponte o viadotto).

E' stata effettuata questa scelta in quanto le entità figlie non erano direttamente coinvolte in nessuna relazione o funzionalità.

La generalizzazione con *Appalto* viene invece risolta lasciando le due entità figlie e correlandole con il padre tramite una relazione.

Questo perché è presente una relazione specifica con l'Appalto Chiuso.

Infine quella con *Utente* viene risolta in modo parziale; collassano nel padre le entità Ministero e Società Autostrada tramite un attributo simile a come avviene su Infrastruttura, rimane l'entità Società Manutenzione tramite una relazione come avviene con Appalto.

Una volta ristrutturato lo schema ER si procede, tramite le regole di mappatura, alla stesura del modello Relazionale.

Seguendo le regole di mappatura è possibile trasformare lo schema ER in modello Relazionale.

La terza regola di mapping è già stata applicata.

Entità:

Parametro (Parametro)

Autostrada (Codice, Lunghezza)

Utente (Email, Password, Categoria)

SocietaManutenzione (*Utente*, DataRegistrazione)

V.I.R. [Utente con Utente.Email]

Infrastruttura (CodiceInfr, Nome, IndiceBonta, Tipo, Latitudine, Longitudine, *Autostrada*)

V.I.R. [Autostrada con Autostrada.Codice]

Sensore (IdSensore, *Infrastruttura*, *Parametro*)

V.I.R. [Infrastruttura con Infrastruttura.CodiceInfr]

V.I.R. [Parametro con Parametro.Parametro]

StoricoRilevazioni (*Sensore*, DataRilevazione, Valore)

V.I.R. [Sensore con Sensore.IdSensore]

Appalto (IdAppalto, DataApertura, *Parametro*, *Infrastruttura*)

V.I.R. [Infrastruttura con Infrastruttura.CodiceInfr]

V.I.R. [Parametro con Parametro.Parametro]

AppaltoAperto (*IdAppalto*)

V.I.R. [IdAppalto con Appalto.IdAppalto]

AppaltoConcluso (*IdAppalto*, DataEsecuzioneIntervento, *SocietaManutenzione*)

V.I.R. [IdAppalto con Appalto.IdAppalto]

V.I.R. [SocietaManutenzione con SocietaManutenzione.Utente]

Relazioni:

Disponibilita (*Parametro*, *SocietaManutenzione*)

V.I.R. [Infrastruttura con Infrastruttura.CodiceInfr]

V.I.R. [Parametro con Parametro.Parametro]

In seguito alla definizione del modello Relazionale verrà implementato tramite SQL.

3.4 Progettazione Fisica

Di seguito verranno descritte alcune query per la definizione della struttura di alcune tabelle.

```
33  CREATE TABLE `Appalto` (
34    `IdAppalto` int(11) NOT NULL,
35    `DataApertura` datetime NOT NULL DEFAULT current_timestamp(),
36    `Parametro` varchar(32) NOT NULL,
37    `Infrastruttura` int(11) NOT NULL
38  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

figura 6

[...]