



4OURSQUARED

4oursquared.unipd@gmail.com

Lumos Minima
Imola Informatica

Specifica Tecnica

Informazioni

<i>Redattori</i>	Soldà Matteo
<i>Versione</i>	0.0.1
<i>Uso</i>	esterno

Descrizione

Questo documento descrive la specifica architettuale del progetto, illustrando gli aspetti progettuali che caratterizzino il design del prodotto.



Versione	Data	Redattore	Verificatore	Descrizione
0.0.1	20/09/2023	Soldà Matteo	Alberti Nicolas	Stesura delle prime sezioni.
0.0.0	05/09/2023	Soldà Matteo	Alberti Nicolas	Prima stesura.



Contents

1	Introduzione	1
1.1	Scopo del Documento	1
1.2	Riferimenti Normativi	1
1.3	Riferimenti Informativi	1
2	Tecnologie Utilizzate	2
2.1	Frontend	2
2.2	Backend	2
2.3	Database	2
2.4	Meccanismo di Comunicazione	2
3	Architettura Logica	3
3.1	Frontend	3
3.2	Backend	3
3.3	Database	3
4	Architettura di Deployment	4
5	Design Pattern	5
6	Idiomi	6
7	Altri Aspetti Progettuali Rilevanti	7
7.1	Comunicazione Automatica	7
7.2	Autenticazione e Autorizzazione tramite JWT	7



List of Figures

1	Diagramma di flusso del segnale relativo al movimento rilevato dai sensori.	7
2	Diagramma di flusso del meccanismo di autorizzazione tramite JWT.	8
3	Diagramma di flusso del meccanismo di autorizzazione tramite middleware.	9



1 Introduzione

1.1 Scopo del Documento

Questo documento ha lo scopo di descrivere nel dettaglio, soprattutto tramite diagrammi, le caratteristiche architetture del prodotto software sviluppato.

1.2 Riferimenti Normativi

- Capitolato C2 - Lumos Minima

1.3 Riferimenti Informativi

- Design Pattern Architetture
- Dependency Injection
- MVC e Derivati
- Pattern Creazionali
- Software Architecture
- Pattern Strutturali
- Pattern Comportamentali
- Programmazione SOLID



2 Tecnologie Utilizzate

2.1 Frontend

Per realizzare il frontend, ossia la parte di applicazione che viene eseguita sul browser dell'utente, sono state utilizzate le seguenti tecnologie:

- React: libreria JavaScript per la creazione di interfacce utente;
- Typescript: linguaggio di programmazione che estende JavaScript aggiungendo i tipi, permettendo una codifica più robusta e sicura;
- Bootstrap: framework CSS per la creazione di interfacce responsive e accattivanti;

2.2 Backend

Per realizzare il backend, ossia la parte di applicazione che viene eseguita sul server, sono state utilizzate le seguenti tecnologie:

- Node.js: runtime JavaScript che permette di eseguire codice JavaScript lato server;
- Typescript: linguaggio di programmazione che estende JavaScript aggiungendo i tipi, permettendo una codifica più robusta e sicura;
- Express: framework che permette di creare applicazioni web e API più facilmente e con una miglior gestione;
- Axios: client HTTP basato su promise per effettuare richieste HTTP basate su Promise;
- Mongoose: libreria che permette di gestire in modo più semplice e intuitivo i database MongoDB;
- Cors: middleware che permette di configurare in modo semplice e veloce le politiche CORS;
- JWT: libreria che permette di gestire in modo semplice e veloce i token JWT;
- Cron: libreria che permette di gestire in modo semplice e veloce i cronjob, ossia per la creazione di routine automatiche;

2.3 Database

Il database utilizzato è di tipo NoSql, in particolare MongoDB, che permette di gestire documenti in formato BSON (Binary JSON).

2.4 Meccanismo di Comunicazione

Tutte le comunicazioni, sia esterne (da client a server) che interne (da server a server) sono state gestite tramite API REST.



3 Architettura Logica

3.1 Frontend

3.2 Backend

3.3 Database



4 Architettura di Deployment



5 Design Pattern



6 Idiomi

7 Altri Aspetti Progettuali Rilevanti

7.1 Comunicazione Automatica

Per quanto riguarda il meccanismo che permette di aumentare e diminuire automaticamente la luminosità dei lampioni che si trovano dentro una determinata area illuminata, il soggetto principale sono i sensori: essi, qualora rilevassero il movimento di una persona, di un veicolo o di un animale, dovrebbero inviare una richiesta di tipo HTTP all'endpoint prestabilito. Alla ricezione della richiesta, il server si occuperà della gestione della stessa come descritto nel diagramma sottostante.

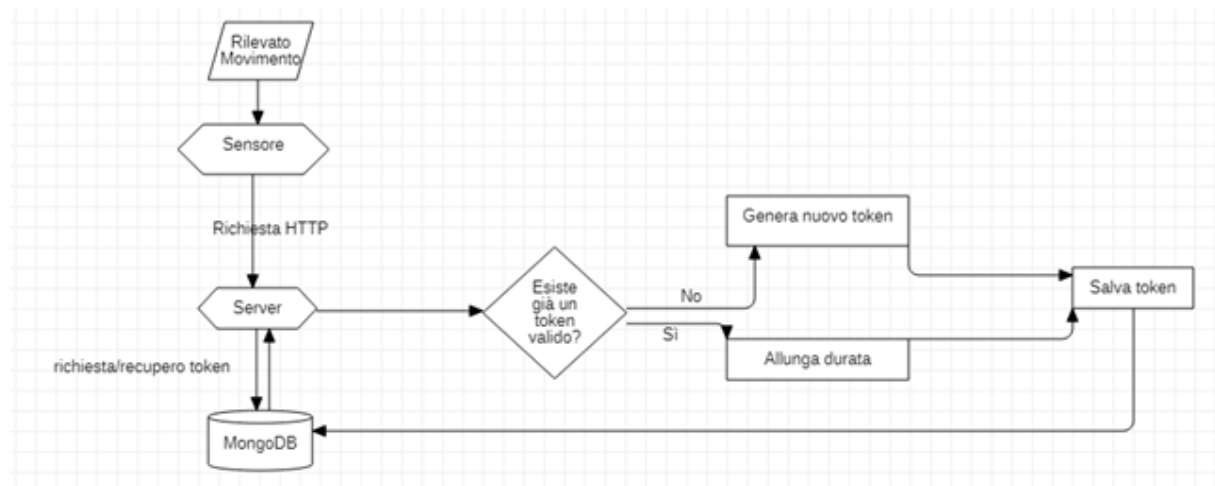


Figure 1: Diagramma di flusso del segnale relativo al movimento rilevato dai sensori.

Il token trova un utilizzo concreto solo per i lampioni dell'area che sono configurati in modalità PULL (manuale).

Per questo motivo, di seguito c'è una piccola spiegazione di come viene gestito il rilevamento di movimento da entrambi i tipi di lampione:

- PUSH: i lampioni si illuminano quando il sensore segnala un movimento, senza necessità di utilizzare il token
- PULL: la parte di server che gestisce l'area illuminata, con cadenza regolare e definita dall'attributo *polling time*, controlla nel database se c'è un token valido per l'area di riferimento:
 - Se il token è valido e inutilizzato: illumina tutti i lampioni impostati in modalità PULL;
 - Se il token non è valido ma è stato utilizzato: riduce la luminosità di tutti i lampioni a quella iniziale;
 - Negli altri casi, verrà restituito un codice stato consono, ma senza effettuare nessun'altra azione.

7.2 Autenticazione e Autorizzazione tramite JWT

Dopo aver inserito le proprie credenziali nella login mask, se la password (criptata con hashing SHA512) coincide con quella dell'utente viene restituito un token JWT firmato dal server, una stringa che contiene informazioni sul ruolo dell'utente. L'autorizzazione (necessaria per l'accesso alle pagine dal lato "client", e per utilizzare le API del server) sfrutta questo meccanismo:

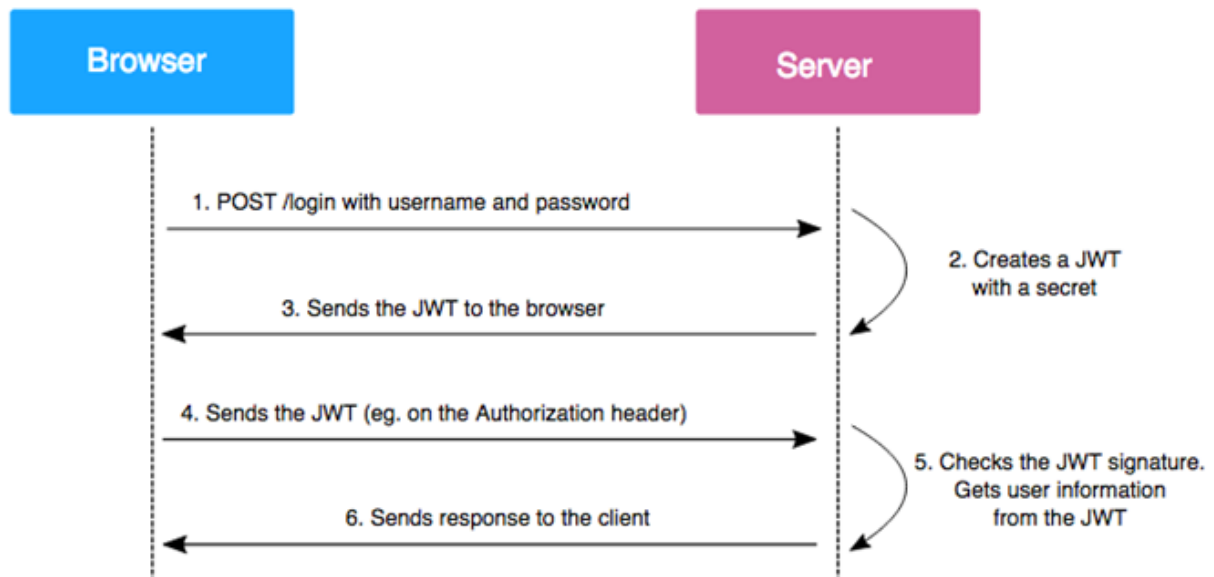


Figure 2: Diagramma di flusso del meccanismo di autorizzazione tramite JWT.

Tuttavia, il JWT è memorizzato (e inviato al server) all'interno di un cookie HTTP-Only: questo impedisce che il contenuto del cookie possa essere letto da codice JavaScript malevolo, vanificando così attacchi alla confidenzialità di tipo XSS (Cross-Site Scripting).

Le API sono protette da due livelli di middleware che prendono in carico la richiesta HTTP.

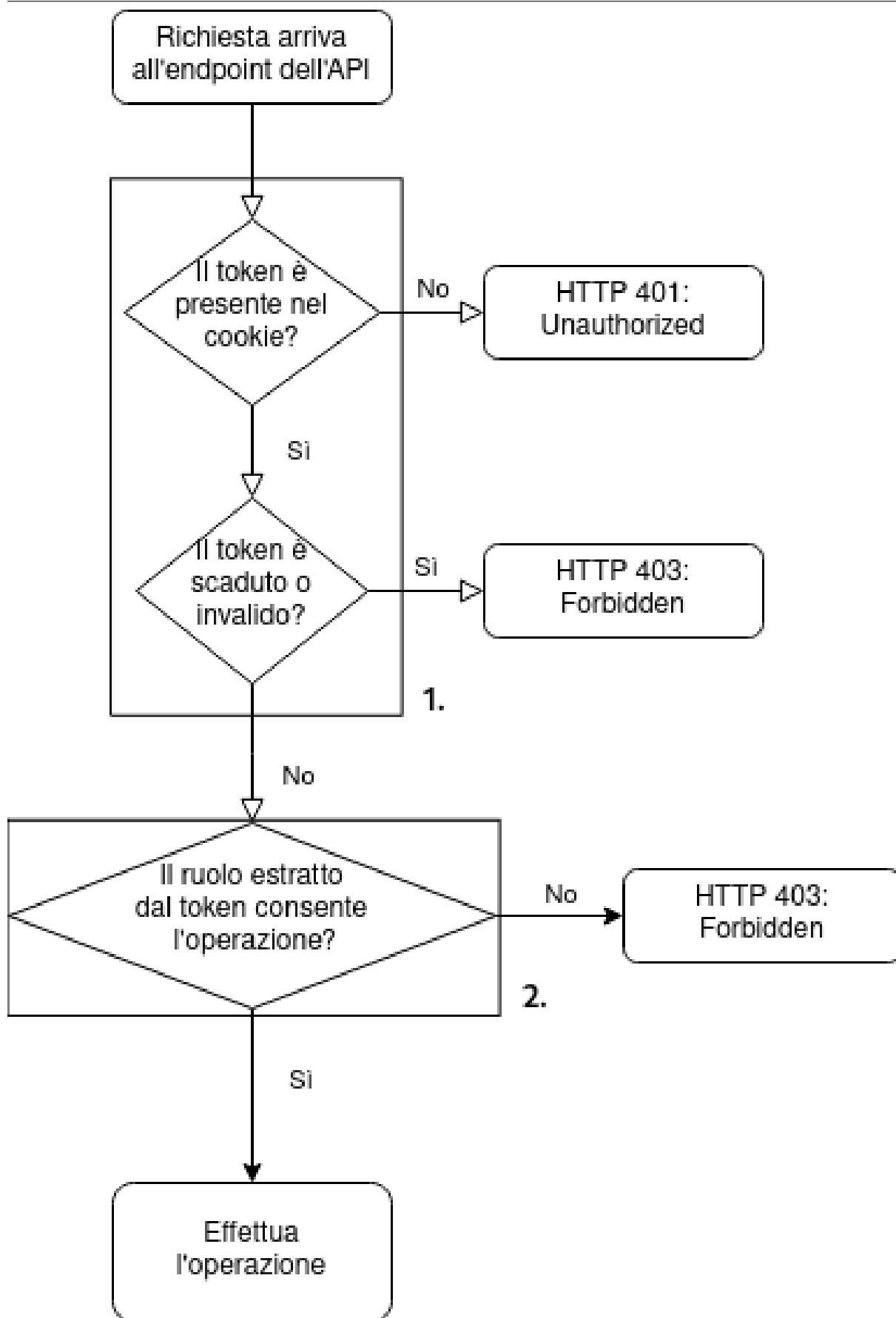


Figure 3: Diagramma di flusso del meccanismo di autorizzazione tramite middleware.