



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA  
CAMPUS DI CESENA

Corso di Laurea Magistrale in  
Ingegneria e Scienze Informatiche

Dipartimento di Informatica,  
Scienze e Ingegneria (DISI)

---

Tesi di Laurea Magistrale in Paradigmi di Programmazione e Sviluppo

# PROCESSO DI TRANSIZIONE E REINGEGNERIZZAZIONE DI UN PANNELLO WEB PER DNS FILTERING

Relatore:

**Prof. Mirko Viroli**

Correlatori:

**Dott. Nicolas Farabegoli**

**Dott. Gianluca Aguzzi**

Sessione di laurea del

14 marzo 2025

(a.a. 2023/2024)

Candidato:

**Alberto Spadoni**



# Indice dei contenuti

## **Argomenti della presentazione:**

- Introduzione e motivazioni
- Background
- Analisi del sistema esistente
- Design del nuovo sistema
- Implementazione
- Valutazione dei risultati
- Conclusioni

# Introduzione

- **Esperienza di tirocinio:** collaborazione con un'azienda specializzata nel filtraggio DNS.
- **Scopo della tesi:** documentare e contribuire al processo di reingegnerizzazione del sistema esistente.
- **Motivazione della reingegnerizzazione:** posizionamento strategico verso medie-grandi imprese attraverso un prodotto con caratteristiche enterprise.
- **Limiti attuali:** scalabilità limitata e UI poco attraente per i clienti *enterprise*.

**Flash**  **Start**®  
INTERNET PROTECTION

# Background

## Teoria alla base del progetto

- **Domain Name System (DNS)**

- Sistema che traduce nomi di dominio in indirizzi IP.
- [www.airbus.com](http://www.airbus.com) → 107.154.75.155

- **Filtraggio DNS**

- Tecnica per bloccare o consentire l'accesso a determinati domini.
- Garantisce sicurezza e controllo della rete.

- **Reingegnerizzazione software**

- Processo fondamentale per modernizzare sistemi esistenti.
- Migliora la manutenibilità, la scalabilità e la sicurezza.

## Analisi del sistema legacy

- **Funzionalità principali:**

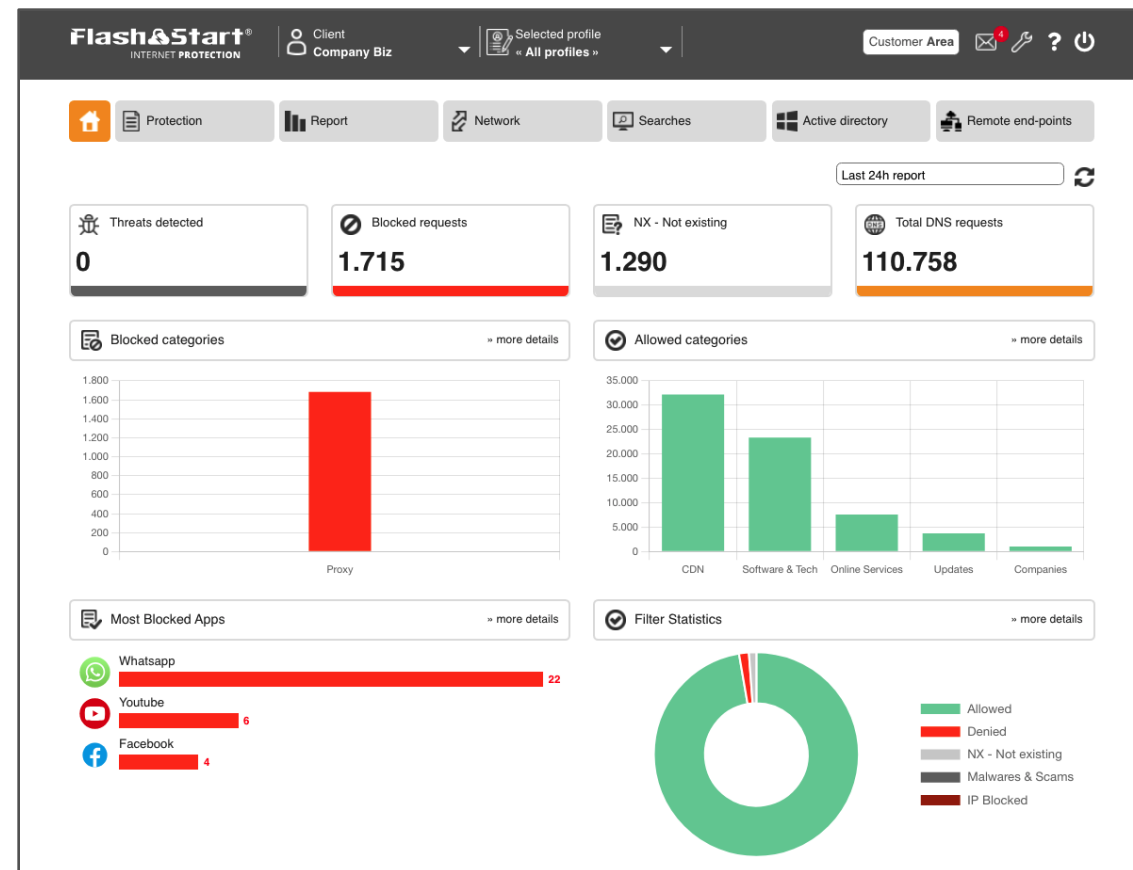
- Configurazione della protezione.
- Gestione dei report.
- Monitoraggio del traffico DNS.

- **Architettura attuale:**

- Monolitica con tecnologie obsolete.
- Scarsa separazione delle responsabilità.

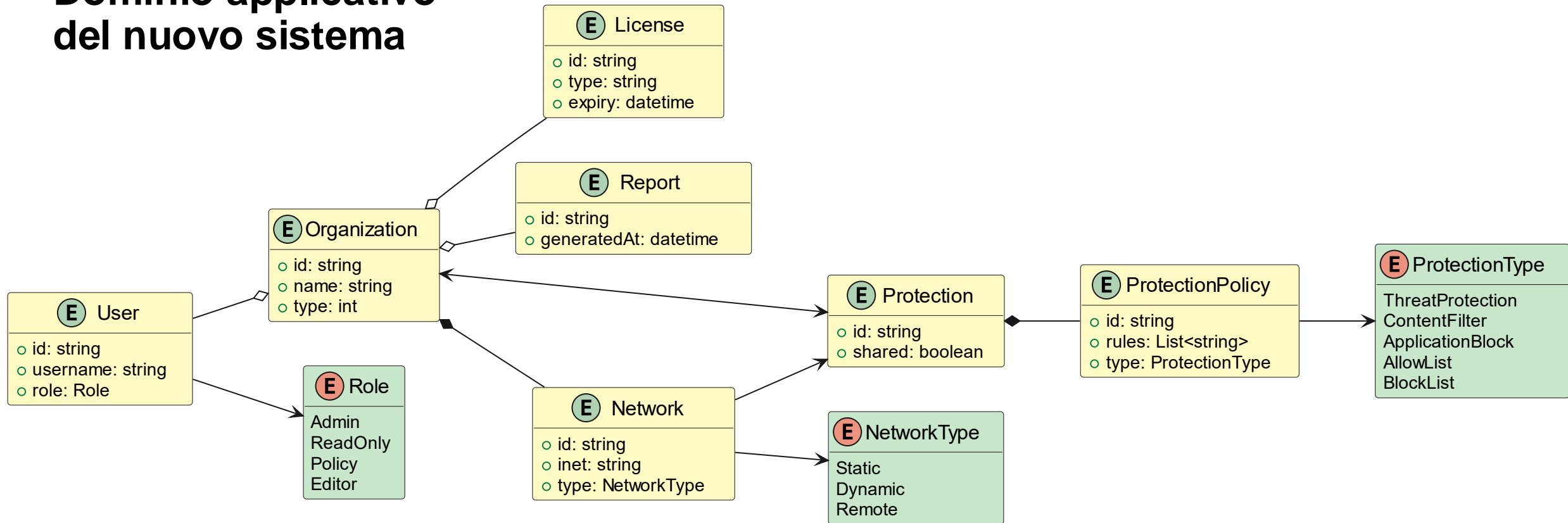
- **Limiti riscontrati:**

- Manutenzione difficile e onerosa
- Mancanza della multiutenza
- UI/UX poco curata



# Analisi (2/2)

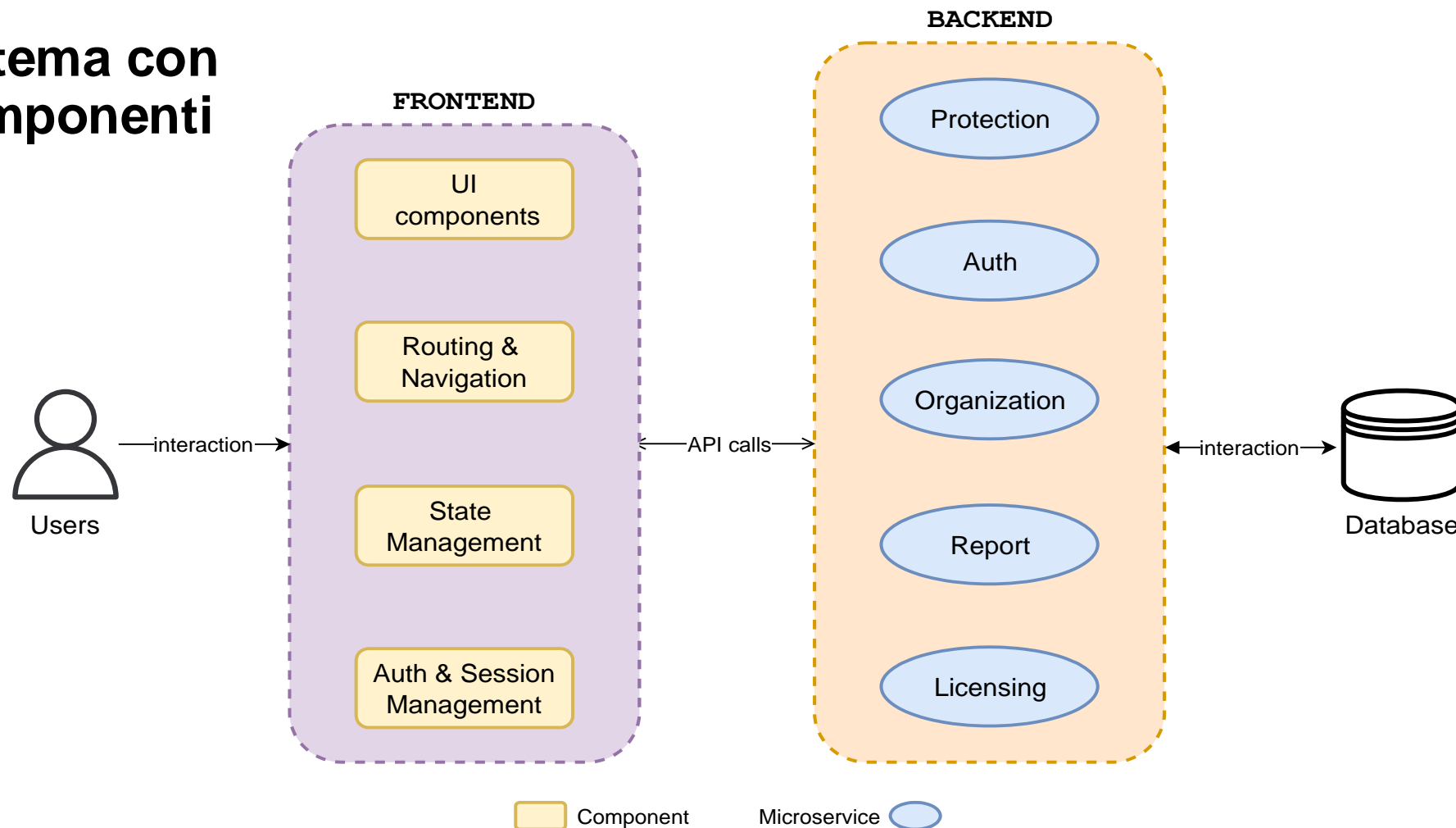
## Dominio applicativo del nuovo sistema



# Design (1/2)

## Architettura del sistema con interazione tra i componenti

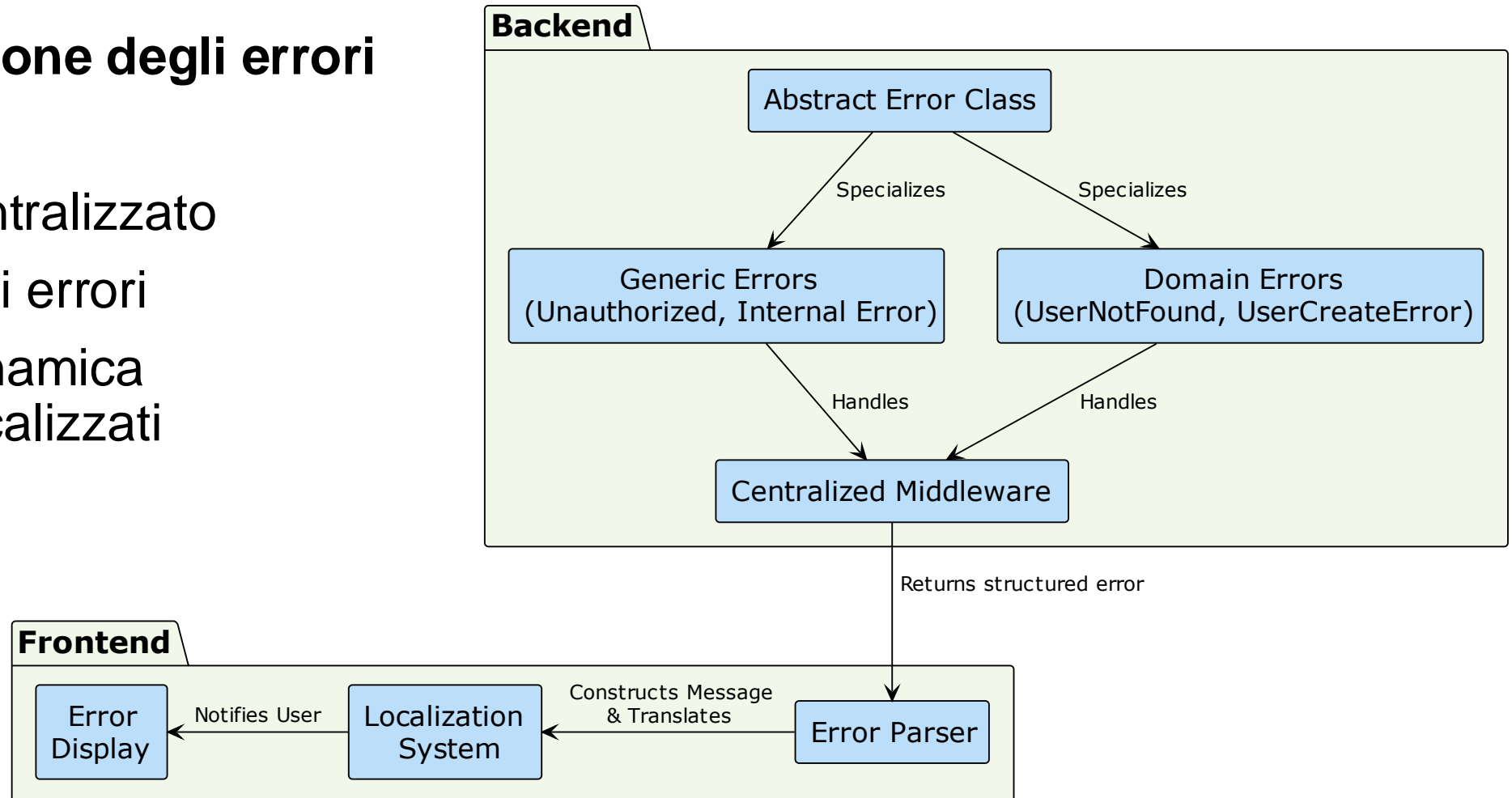
- Modulare e orientata ai microservizi
- Struttura interna a livelli
- Forte separazione delle responsabilità
- Interazione solo per mezzo di HTTP



# Design (2/2)

## Sistema di gestione degli errori

- Classe astratta
- Middleware centralizzato
- Decodifica degli errori
- Costruzione dinamica di messaggi localizzati





# Implementazione (1/2)

## Piano di sviluppo

- **Metodologia**

- Agile ispirata a SCRUM.
- Sprint bisettimanali e meeting di chiusura.

- **Team**

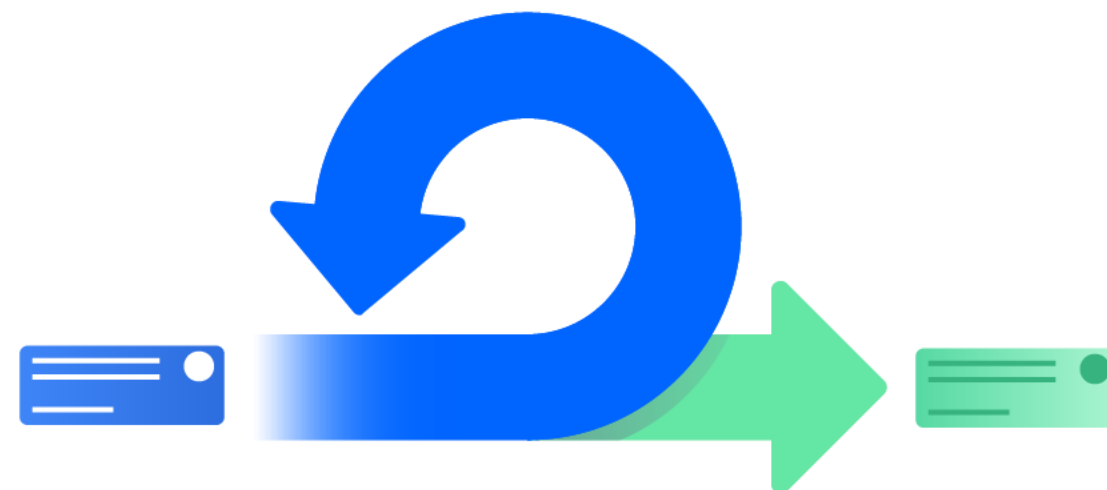
- Full-Stack Developer.
- Design UI/UX affidato a una figura esterna.

- **Repository**

- Monorepo con pipeline CI/CD per build, test e deploy.

- **Reingegnerizzazione**

- Approccio ibrido.
- Big Bang + Evolutivo.



# Implementazione (2/2)

## Implementazione gestione degli errori

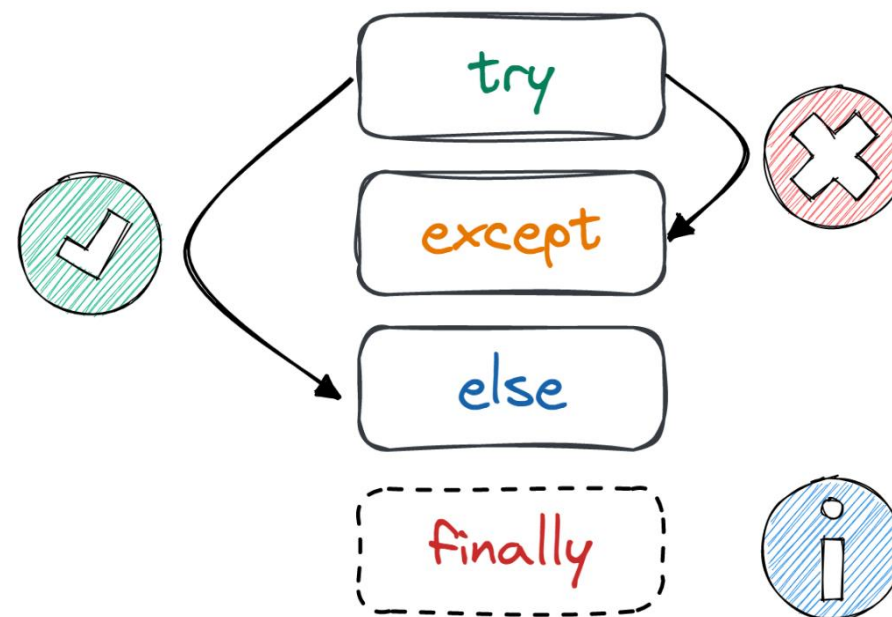
- **Uniformità e robustezza**
  - Errori standardizzati e gestione centralizzata.
- **Esperienza utente migliorata**
  - Messaggi chiari, dinamici e localizzati.
- **Facilità di manutenzione**
  - Modifiche e aggiunte future semplificate dall'approccio modulare.

```
type ErrorDetailsMapping = {  
  "validation.input_error": {  
    entity: ErrorEntity;  
    operation: ErrorOperation;  
    inputErrors: InputError[];  
  };  
  
  "user.not_found": { userId: string };  
  "user.create_failed": {  
    cause: CreateErrorCause;  
  };  
  
  "internal.server_error": { message?: string };  
  "internal.database_error": { message?: string };  
};
```

# Valutazione

## Risultati ottenuti

- **Dashboard panoramica**
  - Visione centralizzata attività clienti per MSP.
- **Multiutenza avanzata**
  - Accessi differenziati per utenti e organizzazioni.
- **Gestione strutturata degli errori**
  - Standardizzazione e robustezza.
- **Supporto multilingua**
  - Predisposto per mercato internazionale.



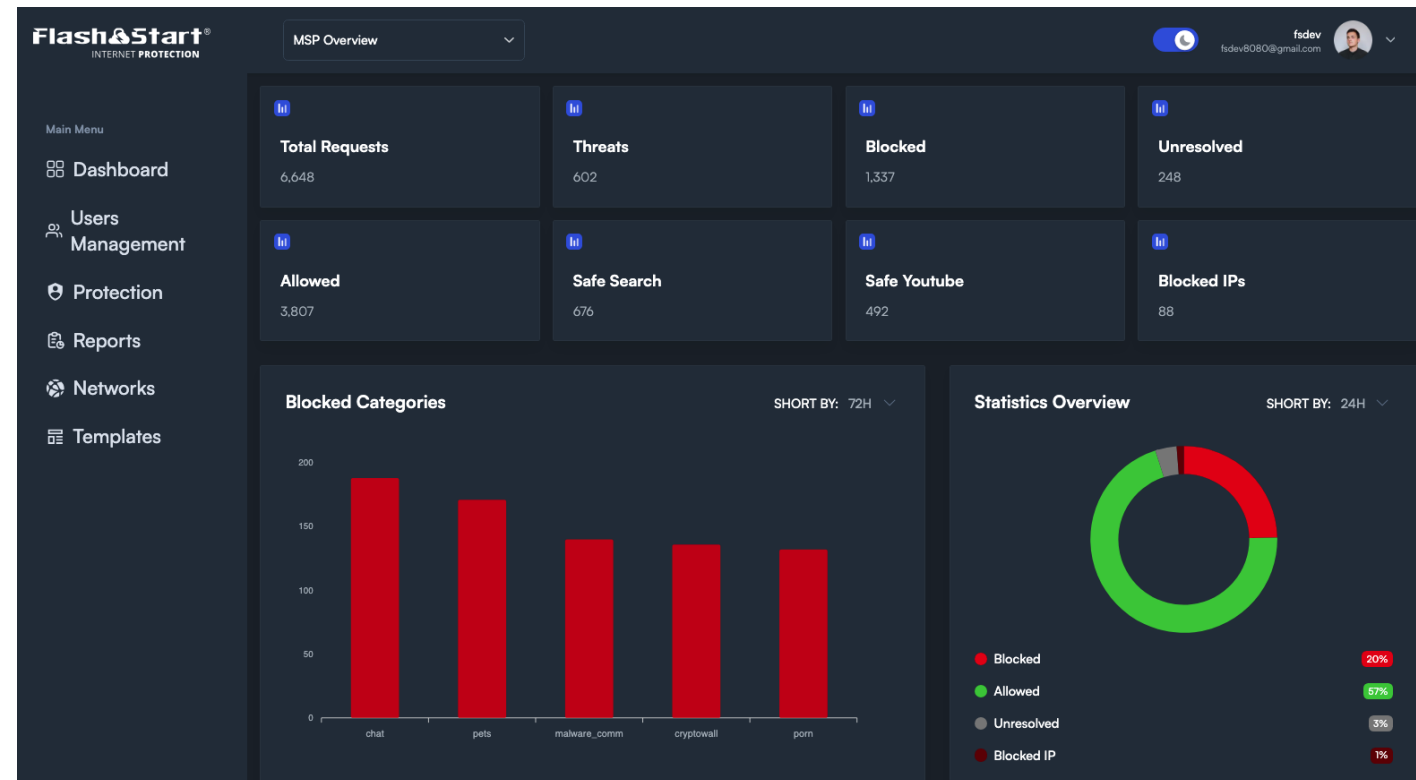
# Conclusione

- **Sintesi del lavoro svolto**

- Progettata un'architettura modulare e scalabile.
- Introdotta la multiutenza e una dashboard per MSP.
- Ridisegnato il database per garantire coerenza e integrità dei dati.

- **Sviluppi futuri**

- Integrazione con Identity Provider esterni.
- Automazione dell'allineamento tra specifiche API e implementazione.





ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA  
CAMPUS DI CESENA



```
const questions: Question[] = await fetchAudienceQuestions(/* no pressure! */);
```