

UNIVERSITÀ DEGLI STUDI DI  
CASSINO E DEL LAZIO MERIDIONALE



DIPARTIMENTO DI INGEGNERIA ELETTRICA  
E DELL'INFORMAZIONE "MAURIZIO SCARANO"

CORSO DI LAUREA IN INGEGNERIA INFORMATICA  
E DELLE TELECOMUNICAZIONI

**Assessment delle performance di Elixir  
nell'ambito IOT.**

**Relatore:**

Prof. Ciro D'Elia

**Candidato:**

Nico Fiorini

ANNO ACCADEMICO 2022/2023

Questa è una dedica

La perfezione non è il nostro obbiettivo ma la nostra tendenza

*Omar Palermo*



# Abstract

L'industria del software si trova a fronteggiare la necessità di sviluppare software sempre più scalabili e performanti per fronteggiare l'aumento degli utenti e di servizi che ne fanno utilizzo. In questo contesto, Elixir, un linguaggio di programmazione funzionale e concorrente basato su Erlang, emerge come una scelta promettente per la costruzione di sistemi altamente affidabili e reattivi, semplificando di molto lo sviluppo di software concorrentiale.

Questo studio si propone di analizzare le caratteristiche di Elixir e le sue performance attraverso una serie di esperimenti empirici esplorando diversi aspetti delle performance mettendo in rilievo vantaggi e svantaggi nell'adottarlo.

I risultati di questa ricerca forniranno una comprensione approfondita delle capacità di Elixir in termini di prestazioni e affidabilità consentendo agli sviluppatori di fare una scelta pensata alle esigenze dei loro progetti.

# Indice

1	Introduzione	1
2	Caratteristiche concorrenziali	3
	Bibliografia	5

# Capitolo 1

## Introduzione

Elixir è un linguaggio sviluppato nel 2012 da José Valim, con l'obiettivo di consentire una maggiore estensibilità e produttività nella macchina virtuale di Erlang, mantenendo al contempo la compatibilità con l'ecosistema di Erlang[1]. Per questo Elixir si è affermato come una scelta promettente nell'industria del software, mettendosi in rilievo proprio dove l'applicazione ha bisogno di essere scalabile, tollerante agli errori, e responsivo grazie all'approccio concorrenziale.

In particolare, Elixir può essere una scelta utile anche nel dominio dell'IoT per vari motivi:

- **Concorrenza:** Nell'IoT la concorrenza e la scalabilità sono fondamentali, proprio in quest'ambito possono esserci numerosi dispositivi che devono essere monitorati e controllati contemporaneamente, e la capacità di gestire facilmente la concorrenza è fondamentale.
- **Fault Tolerance:** Soprattutto controllando dispositivi hardware, dobbiamo tenere in conto che questi possono improvvisamente smettere di funzionare, ed Elixir può essere in grado di controllare processi che vanno in errore facilmente con la supervisione.
- **Sviluppo Rapido e Manutenzione:** Elixir è un linguaggio abbastanza recente, la sintassi è snella e con feature essenziali, fornendo un build tool (Mix) che fornisce la gestione delle dipendenze attraverso il package Manager Hex [2], un ambiente interattivo(iex) e supporta la generazione automatica della documentazione .

Per questo Elixir si è affermato come una scelta promettente nell'industria del software, proprio nell'a Elixir si è affermato come una scelta promettente nell'industria del software grazie ai vantaggi offerti dal paradigma funzionale

e dall'approccio all'immutabilità dei dati. Questa combinazione consente lo sviluppo di software estremamente robusto. In particolare, l'uso dell'immutabilità dei dati in Elixir semplifica significativamente la programmazione concorrente eliminando la necessità di gestire dati condivisi tra processi, il che porta a una maggiore chiarezza e affidabilità nel codice evitando i possibili comportamenti indeterminati che si potrebbero ottenere in un linguaggio basato su stati.

Il codice è compilato per la macchina virtuale Beam, anche detta Erlang VM, quindi un programmatore Elixir può chiamare qualsiasi funzione Erlang senza un costo di overhead durante l'esecuzione, dando ampio spazio alla completa piattaforma OTP di cui Erlang fa uso.[3]

Elixir in particolare è un linguaggio moderno, infatti arriva con molti vantaggi nell'utilizzarlo rispetto ad Erlang, infatti tutto ciò che si può fare con Elixir si può fare anche con Erlang, ma Elixir porta ad un livello superiore la semplicità di espressione del codice attraverso una sintassi snella e comprensiva.

## Capitolo 2

### Caratteristiche concorrenziali



Ciao [dirac]

# Bibliografia

- [1] *Elixir (programming language)* - *Wikipedia*. [https://en.wikipedia.org/wiki/Elixir\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Elixir_(programming_language)). (Accessed on 03/14/2024).
- [2] *Hex*. <https://hex.pm/>. (Accessed on 03/14/2024).
- [3] *Open Telecom Platform* - *Wikipedia*. [https://en.wikipedia.org/wiki/Open\\_Telecom\\_Platform](https://en.wikipedia.org/wiki/Open_Telecom_Platform). (Accessed on 03/14/2024).