

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
	Bibliografia	3

# Capitolo 1

## Introduzione

Elixir è un linguaggio di programmazione dinamico e funzionale 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, alcuni dei quali sono:

1. **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.
2. **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.
3. **Sviluppo Rapido e Manutenzione:** Elixir è un linguaggio moderno, imparando dalla storia ormai i linguaggi moderni forniscono una sintassi efficiente e snella 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.

Il lavoro di tesi vuole mettere in luce i punti di forza di un linguaggio funzionale e come essi vengono sfruttati in Elixir, si vuole mettere in evidenza

anche la semplicità nell'usare più processi, nel fare una scelta la semplicità anche ricopre un ruolo chiave, deve essere adatto e alla portata di qualsiasi programmatore. La semplicità è un punto chiave ma un buon linguaggio deve essere il più efficiente possibile almeno per gli ambiti di interesse, per questo vengono fatti una serie di test empirici per valutare le performance di questo. tutto sulle performance probabilmente non esisterebbe python.

In particolare il lavoro effettuato è così ripartito:

- Nel capitolo 2 si accenna al linguaggio funzionale ed al lambda calcolo, come questo paradigma differisce rispetto ad un linguaggio basato su stati. Si spiegano anche le astrazioni che Elixir fornisce per poter sviluppare codice affidabile
- Nel capitolo 3 si spiega il lavoro sperimentale svolto e i risultati ottenuti

# Bibliografia

- [1] *Elixir (programming language)* - *Wikipedia*. [https://en.wikipedia.org/wiki/Elixir\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Elixir_(programming_language)).
- [2] *Hex*. <https://hex.pm/>.