

# Identification of Module Boundaries in a Modular Monolith Architecture using Automated technologies

UNIVERSITY OF TURKU  
Department of Computing  
Master of Science (Tech) Thesis  
February 2024  
Florian Dejonckheere

UNIVERSITY OF TURKU  
Department of Computing

FLORIAN DEJONCKHEERE: Identification of Module Boundaries in a Modular Monolith Architecture  
using Automated technologies

Master of Science (Tech) Thesis, 4 p., 4 app. p.  
Department of Computing  
February 2024

---

The modular monolith architecture emerged in recent years as the harmonization of the monolithic and microservices architectures. The paradigm offers a compromise between modularity, flexibility, and scalability. Many monolithic applications are being migrated to modular monoliths or microservices entirely, to satisfy increasingly complex and volatile business requirements. This process is labour-intensive, slow, and may take months to years for larger codebases. Modularization of a codebase typically requires the developer to have an intimate knowledge of both the application code and domain.

In this thesis, we investigate the modular monolith software architecture, and how modules are typically determined as part of the modularization efforts. We propose an automated solution based on dependency analysis and machine learning algorithms to aid in the identification of module boundaries, and evaluate its effectiveness using a case study. We discuss the results and draw conclusions about the proposed solution.

**Keywords:** software architecture, monolith, microservices, modular monolith

# Contents

<b>1. Introduction .....</b>	<b>1</b>
1.1. Problem statement .....	1
1.2. Scope and goal .....	1
1.3. Motivation .....	1
1.4. Methodology .....	1
1.5. Outline .....	1
<b>2. Background .....</b>	<b>2</b>
2.1. Monolith architecture .....	2
2.2. Microservice architecture .....	2
<b>3. Related work .....</b>	<b>3</b>
<b>4. Modular monolith architecture .....</b>	<b>4</b>
4.1. Background .....	4
4.2. Challenges and opportunities .....	4
4.3. Modularization .....	4
<b>5. Proposed solution .....</b>	<b>5</b>
<b>6. Case study .....</b>	<b>6</b>
6.1. Background .....	6
6.2. Analysis .....	6
6.3. Evaluation and results .....	6
6.4. Discussion .....	6
<b>7. Conclusion .....</b>	<b>7</b>
7.1. Future work .....	7
<b>References .....</b>	<b>9</b>

## List of Figures

## List of Tables

## List of Acronyms

<b>API</b>	Application Programming Interface
<b>UI</b>	User Interface

# 1. Introduction

## 1.1. Problem statement

## 1.2. Scope and goal

This research is centered around two research questions:

**Research Question 1:** Which challenges and opportunities arise when considering adoption of the modular monolith architecture in an existing codebase?

**Research Question 2:** How can (*automated technology*) effectively identify optimal module boundaries in a modular monolith architecture?

To answer the first research question, we first define the modular monolith architecture, and examine what sets it apart from monolithic and microservices architectures. Then, we proceed to investigate the merits and drawbacks of the software architecture when applied to an existing codebase.

For the second research question, we enumerate the existing technologies to aid modularization of monolithic codebases, and choose one automated technology for further examination. (*Automated technology*) is then implemented for a given use case, and compared to manual modularization efforts in terms of accuracy, efficiency, development velocity.

The goal of this research can be summarized as follows:

1. Investigate the merits and drawbacks of the modular monolith architecture in an existing codebase
2. Investigate the use of automated technologies to modularize a monolithic architecture

## 1.3. Motivation

## 1.4. Methodology

## 1.5. Outline

## **2. Background**

### **2.1. Monolith architecture**

### **2.2. Microservice architecture**



### **3. Related work**

## **4. Modular monolith architecture**

### **4.1. Background**

### **4.2. Challenges and opportunities**

### **4.3. Modularization**

## **5. Proposed solution**

## **6. Case study**

### **6.1. Background**

### **6.2. Analysis**

### **6.3. Evaluation and results**

### **6.4. Discussion**

## **7. Conclusion**

### **7.1. Future work**



## References