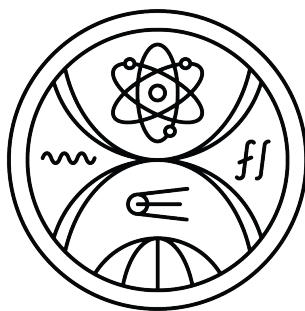


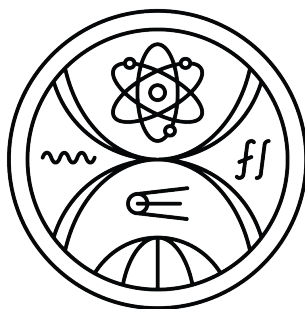
Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky



MODELOVANIE SOFTVÉOVÝCH ŠTÝLOV A VZOROV

Diplomová práca

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky



MODELOVANIE SOFTVÉOVÝCH ŠTÝLOV A VZOROV

Diplomová práca

Študijný program: Aplikovaná informatika
Študijný odbor: Aplikovaná informatika
Školiace pracovisko:: Katedra aplikovanej informatiky
Školiteľ: Ing. Lukáš Radoský
Konzultant:



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Anna Lenhardtová
Študijný program: aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Modelovanie softvérových štýlov a vzorov
Modelling software styles and patterns

Anotácia: Vo vývoji softvéru vznikajú zadania, problémy a situácie, ktoré sa naprieč projektami opakujú. Softvérové inžinierstvo sa takýmito javmi zaoberá a ponúka na nie riešenia. Pre problémy celkovej štruktúry softvérových systémov ponúka architektonické štýly, ktoré na vysokej úrovni abstrakcie definujú štruktúru softvéru. Príkladom sú Pipes&Filters, Blackboard, mikroslužby a podobne. Pre často sa opakujúce problémy na nižšej úrovni granularity ponúka návrhové vzory, napríklad Composite, Abstract Factory a podobne.

Analyzujte možnosti modelovania architektonických štýlov a návrhových vzorov na rovnakej úrovni granularity, napríklad na úrovni diagramu tried. Vytvorte príklady takýchto modelov pre vybrané štýly a vzory. Overte vhodnosť vašich modelov, napríklad ich vykonateľnosťou.

Cieľ: Návrh spôsobu modelovania zvoleného architektonického štýlu či štýlov na nízkej úrovni abstrakcie
Vytvorenie príkladov modelov pre zvolenú množinu štýlov a vzorov

Literatúra: JOUAULT, Frédéric, et al. Designing, animating, and verifying partial UML Models. In: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems. 2020. p. 211-217.

Buschmann F. et al.: Pattern-oriented software architecture: a pattern language for distributed computing, Vol. 4. New York : John Wiley & Sons, 2007.

GAMMA, E.: HELM, R., JOHNSON, R., VLISSIDES, J. (1994), Design Patterns – Elements of reusable object-oriented software code, Addison-Wesley

Richards, M., and Ford, N. 2020. . Fundamentals of Software Architecture: An Engineering Approach. O'Reilly Media, Incorporated.

Poznámka: architektonické štýly, návrhové vzory, softvérové inžinierstvo



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

Vedúci: Ing. Lukáš Radoský
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: doc. RNDr. Tatiana Jajcayová, PhD.
Dátum zadania: 18.11.2024

Dátum schválenia: 21.11.2024

prof. RNDr. Roman Ďurikovič, PhD.
garant študijného programu

.....
študent

.....
vedúci práce

Bratislava, 2026

.....
Bc. Anna Lenhardtová

Pod'akovanie

Abstrakt

Vo vývoji softvérových systémov sa často stretávame s opakujúcimi sa problémami, ktoré vyžadujú overené a efektívne riešenia. Softvérové inžinierstvo na tieto výzvy reaguje prostredníctvom architektonických štýlov a návrhových vzorov, ktoré ponúkajú osvedčené prístupy k organizácii štruktúry a správania systému na rôznych úrovniach abstrakcie. V tejto práci analyzujeme možnosti modelovania vybraných architektonických štýlov a návrhových vzorov — konkrétne štýlu Pipes and Filters a návrhových vzorov Builder a Bridge — na rovnakej úrovni granularity, ako je napríklad diagram tried. Cieľom je vytvoriť zjednotené a vykonateľné modely týchto konceptov v rámci softvérového nástroja AnimArch, ktorý umožňuje vizuálnu animáciu správania systémov. V práci doplníme katalóg systému AnimArch o nové modely a overujeme ich vhodnosť prostredníctvom simulácií a analýzy správania. Výsledkom je metodika, ktorá podporuje návrh a vizualizáciu architektúr, čo prispieva k lepšej pochopiteľnosti, modularite a udržiavateľnosti softvérových riešení.

Kľúčové slová: Architektonické štýly, návrhové vzory, Builder, Pipes & filters

Abstract

In software development, we often encounter recurring problems that require proven and efficient solutions. Software engineering addresses these challenges through architectural styles and design patterns, which offer established approaches to organizing the structure and behavior of a system at different levels of abstraction. This paper analyzes the modeling possibilities of selected architectural styles and design patterns — specifically the Pipes and Filters style and the Builder and Bridge design patterns — at the same level of granularity, such as class diagrams. The aim is to create unified and executable models of these concepts within the AnimArch tool, which allows for the visual animation of system behavior. In this work, we extend the AnimArch system’s catalog with new models and verify their suitability through simulations and behavior analysis. The result is a methodology that supports the design and visualization of architectures, contributing to better comprehensibility, modularity, and maintainability of software solutions.

Keywords: Architecture styles and patterns, Builder, Bridge, Pipes & filters

Obsah

1	UML	2
2	Architektonické vzory a štýly	3
3	Animarch	4
4	Prehľad existujúcich riešení	5
5	Návrh	6
5.1	Vzor Builder (Tvorca, Staviteľ)	6
5.2	Vzor Bridge (Most)	6
5.3	Štýl Pipes & Filters (Rúry/Potrubia a filtre)	6
6	Implementácia	7
6.1	Triedne diagramy	7
6.2	Animácie	7
7	Evaluácia	8

Zoznam obrázkov

Zoznam tabuliek

Terminológia

Termíny

-

Skratky

-

Motivácia

Kapitola 1

UML

Kapitola 2

Architektonické vzory a štýly

Kapitola 3

Animarch

Kapitola 4

Prehľad existujúcich riešení

Kapitola 5

Návrh

5.1 Vzor Builder (Tvorca, Staviteľ)

5.2 Vzor Bridge (Most)

5.3 Štýl Pipes & Filters (Rúry/Potrúbia a filtre)

Kapitola 6

Implementácia

6.1 Triedne diagramy

6.2 Animácie

Kapitola 7

Evalúácia

Záver

Literatúra

- [1] Matej Ferenc, Ivan Polasek, and Juraj Vincur. Collaborative modeling and visualization of software systems using multidimensional uml. In *2017 IEEE Working Conference on Software Visualization (VISSOFT)*, pages 99–103. IEEE, 2017.
- [2] Brent Hailpern and Peri Tarr. Model-driven development: The good, the bad, and the ugly. *IBM systems journal*, 45(3):451–461, 2006.
- [3] Lukas Radosky and Ivan Polasek. Executable multi-layered software models. In *Proceedings of the 1st International Workshop on Designing Software*, pages 46–51, 2024.
- [4] Reinhard Schauer and Rudolf K Keller. Pattern visualization for software comprehension. In *Proceedings. 6th International Workshop on Program Comprehension. IWPC'98 (Cat. No. 98TB100242)*, pages 4–12. IEEE, 1998.
- [5] Anubha Sharma, Manoj Kumar, and Sonali Agarwal. A complete survey on software architectural styles and patterns. *Procedia Computer Science*, 70:16–28, 2015.