

Általános információk, a diplomaterv szerkezete

A diplomaterv szerkezete a BME Villamosmérnöki és Informatikai Karán:

1. Diplomaterv feladatkiírás
2. Címoldal
3. Tartalomjegyzék
4. A diplomatervező nyilatkozata az önálló munkáról és az elektronikus adatok kezeléséről
5. Tartalmi összefoglaló magyarul és angolul
6. Bevezetés: a feladat értelmezése, a tervezés célja, a feladat indokoltsága, a diplomaterv felépítésének rövid összefoglalása
7. A feladatkiírás pontosítása és részletes elemzése
8. Előzmények (irodalomkutatás, hasonló alkotások), az ezekből levonható következtetések
9. A tervezés részletes leírása, a döntési lehetőségek értékelése és a választott megoldások indoklása
10. A megtervezett műszaki alkotás értékelése, kritikai elemzése, továbbfejlesztési lehetőségek
11. Esetleges köszönetnyilvánítások
12. Részletes és pontos irodalomjegyzék
13. Függelék(ek)

Felhasználható a következő oldaltól kezdődő \LaTeX diplomatervsablon dokumentum tartalma.

A diplomaterv szabványos méretű A4-es lapokra kerüljön. Az oldalak tükörmargóval készüljenek (mindenhol 2,5 cm, baloldalon 1 cm-es kötéssel). Az alapértelmezett betűkészlet a 12 pontos Times New Roman, másfeles sorközzel, de ettől kismértékben el lehet térni, ill. más betűtípus használata is megengedett.

Minden oldalon – az első négy szerkezeti elem kivételével – szerepelnie kell az oldalszámnak.

A fejezeteket decimális beosztással kell ellátni. Az ábrákat a megfelelő helyre be kell illeszteni, fejezetenként decimális számmal és kifejező címmel kell ellátni. A fejezeteket decimális alaosztással számozzuk, maximálisan 3 alaosztás mélységben (pl. 2.3.4.1.). Az ábrákat, táblázatokat és képleteket célszerű fejezetenként külön számozni (pl. 2.4. ábra, 4.2. táblázat vagy képletnél (3.2)). A fejezetcímeket igazítsuk balra, a normál szövegnél viszont használjunk sorkiegyenlítést. Az ábrákat, táblázatokat és a hozzájuk tartozó címet igazítsuk középre. A cím a jelölt rész alatt helyezkedjen el.

A képeket lehetőleg rajzoló programmal készítsék el, az egyenleteket egyenlet-szerkesztő segítségével írják le (A \LaTeX ehhez kézenfekvő megoldásokat nyújt).

Az irodalomjegyzék szövegek közötti hivatkozása történhet sorszámozva (ez a preferált megoldás) vagy a Harvard-rendszerben (a szerző és az évszám megadásával). A teljes lista névsor szerinti sorrendben a szöveg végén szerepeljen (sorszámozott irodalmi hivatkozások esetén hivatkozási sorrendben). A szakirodalmi források címeit azonban mindig az eredeti nyelven kell megadni, esetleg zárójelben a fordítással. A listában szereplő valamennyi publikációra hivatkozni kell a szövegben (a \LaTeX -sablon a Bib \TeX segítségével mindezt automatikusan kezeli). Minden publikáció a szerzők után a következő adatok szerepelnek: folyóirat cikkeknél a pontos cím, a folyóirat címe, évfolyam, szám, oldalszám tól-ig. A folyóiratok címét csak akkor rövidítsük, ha azok nagyon közismertek vagy nagyon hosszúak. Internetes hivatkozások megadásakor fontos, hogy az elérési út előtt megadjuk az oldal tulajdonosát és tartalmát (mivel a link egy idő után akár elérhetetlenné is válhat), valamint az elérés időpontját.

Fontos:

- A szakdolgozatkészítő / diplomatervező nyilatkozata (a jelen sablonban szereplő szövegtartalommal) kötelező előírás, Karunkon ennek hiányában a szakdolgozat/diplomaterv nem bírálható és nem védhető!
- Mind a dolgozat, mind a melléklet maximálisan 15 MB méretű lehet!

Jó munkát, sikeres szakdolgozatkészítést, ill. diplomatervezést kívánunk!

FELADATKIÍRÁS

A feladatkiírást a tanszéki adminisztrációban lehet átvenni, és a leadott munkába eredeti, tanszéki pecséttel ellátott és a tanszékvezető által aláírt lapot kell belefűzni (ezen oldal *helyett*, ez az oldal csak útmutatás). Az elektronikusan feltöltött dolgozatban már nem kell beszerkeszteni ezt a feladatkiírást.



Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Measurement and Information Systems

Integration of standard datasources with interactive data visualization solutions

BACHELOR'S THESIS

Author

Márton Orova

Advisor

dr. Zoltán Szatmári
Attila Simon

November 18, 2019

Contents

Kivonat	i
Abstract	ii
1 Introduction	1
1.1 Problem definition	1
1.2 Motivation	1
1.3 Goals	2
2 Background	3
2.1 Datasource	3
2.2 Grafana	3
2.3 JSON	4
2.4 REST API	4
3 Design	5
3.1 Architecture	5
3.2 Components	5
4 Implementation	6
4.1 Gateway	6
4.2 Pros	6
4.3 Cons	6
5 Evaluation	7
6 Future work	8
7 Related works	9
8 Summary	10

Acknowledgements	11
Bibliography	12

HALLGATÓI NYILATKOZAT

Alulírott *Orova Márton*, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző(k), cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy autentikált felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Budapest, 2019. november 18.

Orova Márton
hallgató

Kivonat

Jelen dokumentum egy diplomaterv sablon, amely formai keretet ad a BME Villamosmérnöki és Informatikai Karán végző hallgatók által elkészítendő szakdolgozatnak és diplomatervnek. A sablon használata opcionális. Ez a sablon \LaTeX alapú, a *TeXLive* \TeX -implementációval és a PDF- \LaTeX fordítóval működőképes.

Abstract

This document is a L^AT_EX-based skeleton for BSc/MSc theses of students at the Electrical Engineering and Informatics Faculty, Budapest University of Technology and Economics. The usage of this skeleton is optional. It has been tested with the *TeXLive* T_EX implementation, and it requires the PDF-L^AT_EX compiler.

Chapter 1

Introduction

A bevezető tartalmazza a diplomaterv-kírás elemzését, történelmi előzményeit, a feladat indokoltságát (a motiváció leírását), az eddigi megoldásokat, és ennek tükrében a hallgató megoldásának összefoglalását. A bevezető szokás szerint a diplomaterv felépítésével záródik, azaz annak rövid leírásával, hogy melyik fejezet mivel foglalkozik.

Nowadays, the question of storing, processing and displaying the data is becoming more and more important throughout every industry. The time, when the collected data was only useful for computers and specialists, passed. Today, the need for showing the data in an easily understandable form is significant. It is no wonder, people through the whole hierarchy of a company would like to be well-informed about the results and the ongoing processes. In addition, it is getting highly valuable to be able to display vast amount of data in a way that even outsiders can comprehend.

Because of this trend, many technologies attempting to solve these problems have appeared, creating a wide variety of tools which organizations can use.

In enterprise-grade environments, the use of complex systems - so called data-pipelines - are becoming increasingly common. These tools provide an integrated solution for transforming and querying data coming from datasources built with different technologies. With the help of these data-pipelines, it is possible to collect many types of data, no matter the format or the frequency. All these things for one reason, to prepare the data for machine or human decision-making.

1.1 Problem definition

- many types of datasources, many ways of customizing them -> integration challenges
- standards made by the industry (data formats, accessing data, visualization) -> permeability is not easy

1.2 Motivation

- one visualization tool (Grafana) for multiple datasources in one place (one consistent way of visualizing data)
- open-source development

- integration task -> get familiar with many new technologies
- Grafana: de facto open-source visualization tool

1.3 Goals

- creating a data-gateway for accessing and visualizing data
- using different datasources (different technologies, data formats)
- connecting the gateway to Grafana (industry standard for opensource data visualization)
- presenting the main datasources and their features
 - relational databases
 - time-series databases
 - key-value stores
- discovering available options for interactivity in Grafana
- design a data-gateway for connecting (two-way, duplex) different types of datasources to grafana
- implement a POC data-gateway for connecting a Python based and a RapidMiner based datasource
- present the advantages and disadvantages of the created gateway

Chapter 2

Background

2.1 Datasource

- data formats
 - JSON
 - XML
 - other??
- datasources and their features
 - relational db
 - timeseries
 - key-value
- with examples!

2.2 Grafana

- what is Grafana?
 - dashboard
 - panel
 - datasource
 - user management
 - queries
- connects with many main datasources
- customizable, can write own plugins (panel, datasource, app)
- interactivity features
 - variables
 - setting the time-range
 - customizing builtin Graph Panel
 - data-links in Graph Panel
 - drill-down links

2.3 JSON

- JavaScript Object Notation
- lightweight format for storing and transporting data (w3school)
- often used when data is sent from a server to a web page (w3school)
- "self-describing" and easy to understand
- syntax rules (need good source, currently w3school)
 - data is in name/value pairs
 - data is separated by commas
 - curly braces hold objects
 - square brackets hold arrays
- example

2.4 REST API

Chapter 3

Design

3.1 Architecture

insert architecture diagram

- why do we need a gateway
- how can we access data from RapidMiner WebService
- why is it good to have a python compoment between Grafana and MySQL
 - we can customize it better, what we see from the database
 - can implement business logic, only see business-relevant projections, granularity of the data
 - can aggregate data from different backends
 - can aggregate data with outsider APIs (POC implementation for this use-case)

3.2 Components

- Responsibilities
- Interfaces?
- Grafana
- proxy/gateway
- python-datasource
 - python-datasource
 - mysql
 - weather-api
- RapidMiner stack
 - rapidminer-server
 - job-agent

- database
- Grafana datasource plugin
- Grafana panel plugin

Chapter 4

Implementation

4.1 Gateway

4.2 Pros

4.3 Cons

Chapter 5

Evaluation

Chapter 6

Future work

- Integrate proxy and Grafana RapidMiner datasource into one

Chapter 7

Related works

interactive-piechart-diagram see notebook

Chapter 8

Summary

Acknowledgements

Ez nem kötelező, akár törölhető is. Ha a szerző szükségét érzi, itt lehet köszönetet nyilvánítani azoknak, akik hozzájárultak munkájukkal ahhoz, hogy a hallgató a szakdolgozatban vagy diplomamunkában leírt feladatokat sikeresen elvégezze. A konzulensnek való köszönetnyilvánítás sem kötelező, a konzulensnek hivatalosan is dolga, hogy a hallgatót konzultálja.

Bibliography

- [1] Budapesti Műszaki és Gazdaságtudományi Egyetem Villamosmérnöki és Informatikai Kar. Diplomaterv portál (2011. február 26.). <http://diplomaterv.vik.bme.hu/>.
- [2] James C. Candy. Decimation for sigma delta modulation. 34(1):72–76, 01 1986. DOI: 10.1109/TCOM.1986.1096432.
- [3] Gábor Jeney. Hogyan néz ki egy igényes dokumentum? Néhány szóban az alapvető tipográfiai szabályokról, 2014. <http://www.mcl.hu/~jeneyg/kinezet.pdf>.
- [4] Peter Kiss. Adaptive digital compensation of analog circuit imperfections for cascaded delta-sigma analog-to-digital converters, 04 2000.
- [5] Wai L. Lee and Charles G. Sodini. A topology for higher order interpolative coders. In *Proc. of the IEEE International Symposium on Circuits and Systems*, pages 459–462, 05 4–7 1987.
- [6] Alexey Mkrtychev. Models for the logic of proofs. In Sergei Adian and Anil Nerode, editors, *Logical Foundations of Computer Science*, volume 1234 of *Lecture Notes in Computer Science*, pages 266–275. Springer Berlin Heidelberg, 1997. ISBN 978-3-540-63045-6. DOI: 10.1007/3-540-63045-7_27. URL http://dx.doi.org/10.1007/3-540-63045-7_27.
- [7] Richard Schreier. *The Delta-Sigma Toolbox v5.2*. Oregon State University, 01 2000. <http://www.mathworks.com/matlabcentral/fileexchange/>.
- [8] Ferenc Wettl, Gyula Mayer, and Péter Szabó. *L^AT_EX kézikönyv*. Panem Könyvkiadó, 2004.