

plain .

. ■

.

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 (min-denhol 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, fejeze-tenként decimális számmal és kifejező címmel kell ellátni. A fejezeteket decimális aláosztással számozzuk, maximálisan 3 aláosztás mélységben (pl. 2.3.4.1.). Az ábrákat, táblázatokat és képleteket célszerű fejeze-tenként 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 szö-veg végén szerepeljen (sorszámozott irodalmi hivatkozások esetén hivatkozási sorrendben). A szakirodalmi források címeit azonban mindig az eredeti nyelven kell megadni, esetleg zárójelben a fordítással. A listá-ban 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: folyó-irat 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 szövegtartalom-mal) 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 beleszerkeszteni ezt a feladatkiírást.



Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Hálózati rendszerek és szolgáltatások Tanszék

Identitás információk kezelése biztonsági események kiértékelésében

DIPLOMATERV

Készítette
Bulla Ádám

Konzulens
dr. Czap László
dr. Buttyán Levente

2017. május 10.

Tartalomjegyzék

Kivonat	i
Abstract	ii
1. Bevezetés	1
1.1. A dolgozat célja és felépítése	1
1.2. Security Information and Event Management	1
1.3. Identity management	2
1.4. A feladat specifikálása	2
2. Irodalomkutatás és a felhasznált technológiák	4
2.1. A felhasznált technológiák ismertetése	4
2.1.1. IBM Security QRadar SIEM	4
2.1.2. IBM Security Identity Manager - ISIM	6
2.1.3. Tivoli Directory Integrator - TDI	6
Köszönetnyilvánítás	7
Irodalomjegyzék	8
Függelék	9
F.1. A TeXstudio felülete	9
F.2. Válasz az „Élet, a világmindenség, meg minden” kérdésére	10

HALLGATÓI NYILATKOZAT

Alulírott *Bulla Ádám*, szigorló hallgató kijelentem, hogy ezt a szakdolgozatot/ diplomatervet **(nem kívánt törlendő)** 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, 2017. május 10.

Bulla Ádám
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.

1. fejezet

Bevezetés

1.1. A dolgozat célja és felépítése



A modern informatika egyik  fontosabb szektora az IT Security, amely a számítógép ipar fejlődésével egyre nagyobb szerepet kap. Ahogy a gépek számító kapacitása növekszik, egyre könnyebben megoldhatók olyan problémák, amelyek addig lehetetlennek, elfogadható időben kivitelezhetetlennek tűntek. Ez a fejlődés az egész szektort arra készíti, hogy folyamatosan fejlődjön, a meglévő alkalmazásokat, módszereket, algoritmusokat javítsa. Emellett a modern világban egyre nagyobb vállalatok jönnek létre, amelyeknek egyre nagyobb személyzetre van szükségük a működéshez, ami  szükségessé teszi egy megfelelően stabil és jól kezelhető informatikai támogató réteg kialakítását.


Több ezer, akár több tízezer alkalmazott mellett gyorsan átlathatatlanná válik, hogy kinek milyen eszközökhöz, akár hardveres, akár szoftvereshez van hozzáférése, ezek egy-egyével való beállítása és karbantartása pedig  teljesen lehetetlen a fent említett támogató szoftverek nélkül. Jelen dolgozat az IT Security világának számos területéből kettővel foglalkozik, ennek a kettőnek is elsősorban a kapcsolatával. Ez a két terület a Security Information and Event Management (SIEM), valamint az Identity management (IdM).

A dolgozat felépítése:

- Az 1. fejezet a dolgozat valamint a projekt célját definiálja és járja körbe, gyors bemutatót adva a felhasznált technológiák főbb tulajdonságairól.
- A 2

1.2. Security Information and Event Management

A Security Information and Event Management az informatikai rendszer részeinek monito- ásával foglalkozik biztonsági szempontból. Az infrastruktúrához hozzáfér egy szerver, amelyen egy SIEM szoftver fut, ez végzi a az adatok feldolgozását. Ehhez a szerverhez kapcsolódnak kliensek, amelyek működésükkel kapcsolatos információkat biztosítanak a szerver irányába valamilyen formában, általában log sorokként. A SIEM szerver ezeket feldolgozza, és úgynevezett biztonsági eseményeket hoz belőlük létre, akár egyéb forrásokból érkező információk felhasználásával. Ilyen egyéb forrás lehet  shonnan érkező log sor, hálózati forgalom, valamilyen adatbázisból lekért adatok, vagy előre definiált, a szerverre feltöltött adatok.

Egy esemény jelképez egy olyan történést, amely az operátor számára információval bírhat.  ílván ilyen eseményből egy ember számára feldolgozhatatlan mennyiségű érkezik, főleg egy nagy infrastruktúra esetén, ezért a SIEM megoldások támogatnak valamilyen szabályrendszert, amellyel meg lehet adni az egyes események feldolgozásának

idjét. Ezt általában arra használják, hogy bizonyos paraméterek alapján besorolják az eseményeket bizonyos kategóriákba, többnyire a típusuk és a súlyosságuk alapján, így az operátornak már egy előre szűrt halmazt kell csak végignéznie, valamint ez alapján könnyebb az események látványos kivezetése a felhasználói felületre is.

1.3. Identity management

Az Identity management a fejlődő nagyvállalatok fent említett problémáiból a nagyszámú alkalmazott informatikai hozzáféréseinek kezelését és keretrendszerbeli integrációjának problémáját oldja meg. Az Identity management szoftverek fő feladata, hogy a cég dolgozóit, mint entitásokat, rendszerezzi, csoportokhoz rendeli, és az egyes entitások saját, valamint örökölt jogait érvényre juttatja.

Minden alkalmazotthoz tartozik egy rekord, amely leírja az adott ember személyes adatait és egyéb hasznos információkat, amelyek szükségesek a rendszer üzemeltetéséhez. Ezt a létrejött entitást beosztja a megadott információk szerint a megfelelő, előre definiált csoportokba, amely alapján az jogokat kap bizonyos eszközök használatára. Ezen eszközök is entitásként vannak felvéve a rendszerbe, oly módon, hogy elérhetők az eszközökhöz (akár szoftveres akár hardveres) tartozó információk és használható egy interfész is, amelyen keresztül a felhasználói fiókok módosíthatók, vagy épp hozhatók létre.

1.4. A feladat specifikálása

Ezen dolgozat témája egy valós projekt az IBM magyarországi security services részlegénél, amely a megrendelő által kért verzióon túl egy saját, más alapokon nyugvó változat fejlesztését is célozza. A projekt célja egy integrációs modul fejlesztése, amely képes az IBM által forgalmazott identity manager program (?? ISIM) adatait felhasználni, és az IBM SIEM megoldásának (2.1.1 QRadar) feltölteni a régi felhasználatára.

TODO ! ábra az egész architektúráról

A megrendelő által kért termék a már fent említett integrációs modul elkészítése egy általuk használt másik IBM-es termék, a Tivoli Directory Integrator (TDI) segítségével. A TDI lehetőséget ad egy lánc létrehozására, amely során egy vagy több forrásból adatokat olvasunk be, ezekkel az adatokkal műveleteket végzünk, majd ezt egy kimeneten publikáljuk. Azt az elemet ami a kapcsolatot kezeli egy ki- vagy bemeneti adatforrással connectornak nevezzük. Több beépített connector állt rendelkezésre ehhez az alkalmazáshoz, amelyből használtunk párat, de mivel a kimeneti oldalon a QRadar egy interfésze áll, szükséges volt ehhez egy connector-t fejleszteni. Emellett a projekt ezen része magába foglalta néhány use-case elkészítését is a rendszer felhasználásával. Ilyen use-case például:

TODO ! ábra a TDI-s rendszerről?

- Inaktív személyekhez tartozó felhasználói fiókok
- Valid alkalmazott entitással nem rendelkező felhasználói fiókok
- Adott erőforráshoz hozzáféréssel rendelkező személyek

A megrendelő által kért verzió mellett egy más alapokon működő megoldást is készítettünk, amely nem hagyatkozik a TDI-re, hanem helyette egy Java web applikációként fut. A futtatáshoz az IBM WebSphere nevű keretrendszerét használjuk, amely támogatja a menedzselte futtatást és magasszintű személyre szabási opciókat kínál az egyes feladatok testreszabásához. Emellett biztosít egy webes konténert is, amelyről elérhető az alkalmazás konfigurációs felülete. Ez a megoldás első sorban abban különbözik a TDI-t használótól, hogy míg abban az alkalmazás által biztosított grafikus fejlesztői felület

segítségével kell megadni az egyes feladatokat, a felület minden limitációjával együtt, a WebSphere-t használó esetben a lekérdezések megadhatók a webes grafikus felületen, valamint feltölthetők egy Java osztályként, Jar formátumban. A programnak implementálnia kell a megfelelő interfészt, és ezen keresztül az alkalmazás képes lesz futtanni azt. Ebben a módban tetszőleges működés megvalósítható, a WebSphere által nyújtott korlátokon belül.

2. fejezet

Irodalomkutatás és a felhasznált technológiák

2.1. A felhasznált technológiák ismertetése



Mivel a feladat egy specifikus **termék** előállítása volt, amely már létező termékek közötti kommunikációt biztosít, ezért ennek jelentős része volt a termékekkel való alapszintű, valamint a felhasznált specifikus funkciókkal és interfészekkel való mélyebb ismerkedés.

2.1.1. IBM Security QRadar SIEM


Az IBM Security QRadar SIEM az IBM **hálózati biztonsági platformja**, ami lehetővé teszi eszközöknek és adatoknak a fokozott fenyegetésektől való **védelmét**. A hálózaton elosztott több ezernyi eszközvégpontból és alkalmazásból származó **adatláforrások** eseményadatait összesíti, és a nyers adatokon azonnali normalizálási és összesítési műveleteket végez, ezáltal képes megkülönböztetni a valódi fenyegetéseket a téves riasztásoktól. Az eseménynaplók betöltésére számtalan automatikus módszer áll rendelkezésre, többek közt olyan közismert protokollok mint a SYSLOG, SNMP, FTP, SCP. Az IBM Security QRadar SIEM ugyancsak képes a rendszer sebezhetőségeinek és az esemény- és hálózati adatoknak az összevetésére, ezáltal segítséget nyújt a biztonsági incidensek rangsorolásában. Emellett lehetőség van egyéb adatforrások felvételére a felhasználó által is, amelyek szintén használhatók a fenyegetések és az incidensek detektálásában. Ezek jelentősége elsősorban a dinamikus szabályok létrehozásában játszik nagy szerepet, mivel ezek segítségével egy automatizált programmal mindig a lehető legfrissebben tarthatók az adatok.




2.1. ábra

 pvető működése a szabály alapú szűrésben rejlik. Felvehetők a SIEM-be bizonyos sablonok alapján  zeállítható szabályok, amelyeket a rule engine kiértékel a biztonsági eseményekre. A kiértékelés alapján az eseményeket besorolja a megfelelő csoportokba súlyosságuk és egyéb tulajdonságaik alapján, vagy ha szükséges létrehoz egy új, különálló eseményt. Ha az esemény megfelelően súlyos besorolást kap, akkor a felhasználói felületre kikerül egy értesítés erről, de ezen túl minden feldolgozott esemény később megtekinthető keresések és szűrések segítségével.

A SIEM által kiértékelte eseményekhez egyéb információkat is rendel a rendszer, olyanokat, mint például a támadás típusa, az esemény leírása, a résztvevő felek adatai, melyeket később is meg lehet tekinteni, valamint akár segítségükkel és az egyéb környezeti forgalommal együtt egy egész hálózat működése visszajátszható. A plusz információk közé tartozik egy 3 szempontú értékelés is, amely alapján a QRadar számolja az egyes esetek súlyosságát.

 Súlyosság (Severity) - a súlyosság jelzi hogy a támadó milyen fenyegetettséget jelent annak függvényében hogy mennyire sebezhető a megcélzott eszköz.

- Hitelesség (Credibility) – a hitelesség jelzi az integritását vagy valódiságát egy támadásnak, amelyik érték az eseményt generáló biztonsági eszköztől érkezik. A hitelesség növekedhet, ha a különböző források is jelentik ugyanazt az eseményt.
- Relevancia (Relevance) – a relevancia meghatározza azt, hogy a célpont milyen értéket képvisel a hálózaton.

Ezen dokumentum és feladat szempontjából a legfontosabb része a QRadarnak a dinamikusan feltölthető adathalmazok és azok használata dinamikus szabályokban. Ezek ugyanis  rhetőek egy REST API-n keresztül, így könnyen hozzájuk lehet férni, és így módosítani a szabályok érvényességi körét. Négy féle ilyen adathalmaz áll rendelkezésre, ezek összefoglaló neve a reference data:

- Reference set - Olyan adathalmaz, melyben egyedi értékek sorozata található.
- Reference map - Olyan adathalmaz, melyben kulcs-érték párok találhatók, a kulcsok egyediek, és szigorúan szöveges adatok.
- Reference map of sets - Olyan adathalmaz, melyben kulcs-halmaz párok találhatók, a kulcsok egyediek, szövegesek, és a halmazban saját csoportjukban egyedi értékek találhatók.
- Reference map of maps (tables) - Olyan adathalmaz, melyben kulcs-kulcs-érték triplet összerendelések találhatók.

Minden reference data-nak van egy típusa, ami meghatározza hogy az adott halmazban milyen típusú értékek találhatók.


- ALN - Alfabetikus karakterek
- ALNIC - Alfabetikus karakterek, figyelmen kívül hagyva a kis- és nagybetű közti különbséget
- IP - IP címek
- NUM - Numerikus karakterek
- PORT - Port számok

 DATE - Dátumok, miliszekundumokban 1970.01.01 óta

2.1.2. IBM Security Identity Manager - ISIM

2.1.3. Tivoli Directory Integrator - TDI

A Tivoli Directory Integrator egy általános célú integrációs eszköz, ami lehetővé teszi több, különböző adatforrás koordinálását és integrációját. Mivel a legtöbb forrás más formátumot használ, és máshogy tárolja az adatot, egy ilyen integrációs lépés során szükséges bizonyos átalakításokat elvégezni az adatok között, valamint lehetséges hogy egyéb, plusz lépéseket is szükséges bevezetni, akár más adatforrások bevonásával. Ennek a procedúrának ad keretet a TDI egy grafikus fejlesztő felülettel, valamint a megfelelő Java alapú interfészekkel és kötésekkel, amelyek könnyűvé teszik új komponensek fejlesztését.

A TDI alapvető struktúrája úgynevezett assembly line-okból áll. Egy ilyen assembly line jelképez egy adat transzfert, a kezdeti adatok felolvasásától az átalakításokon át, a végső kimenet feltöltéséig. A ki- és bemeneti interakció ún. connectorokon keresztül történik, amelyek egy egységes interfészt implementálnak, és valamilyen külső adatforráshoz való kapcsolódást valósítanak meg. Mivel a legtöbb adatforrás  s formátumban tárolja az adatokat, a TDI minden be- valamint kimeneti műveletnél biztosít egy hozzárendelési lépést, amellyel megadhatjuk, hogy a külső attribútumok milyen belső attribútumokra legyenek leképezve. Ilyen ún. mapping lépést az assembly line-on bármikor végrehajthatunk, és emellett még számtalan átalakítási lépés áll rendelkezésre, mint például ciklusok vagy elágazások használata. A TDI talán egyik legfontosabb képessége a Javascriptből való testreszabhatóság. Ez azt jelenti, hogy az assembly line-on az adatokat szabadon manipulálhatjuk Javascriptes kódból, létrehozhatunk szkripteket amik az futtatás bizonyos pontjain aktiválódnak, valamint számtalan egyéb funkciót érhetünk el ezekből a programokból, mint a logolás, paraméterek módosítása, vagy arbitrális kód futtatása.

A dolgozat szempontjából az egyik legfontosabb része a TDI-nak a connectorok, mivel a feladat része volt egy ilyen fejlesztése, ami támogatja a kommunikációt egy QRadar szerverrel

Köszönetnyilvánítás

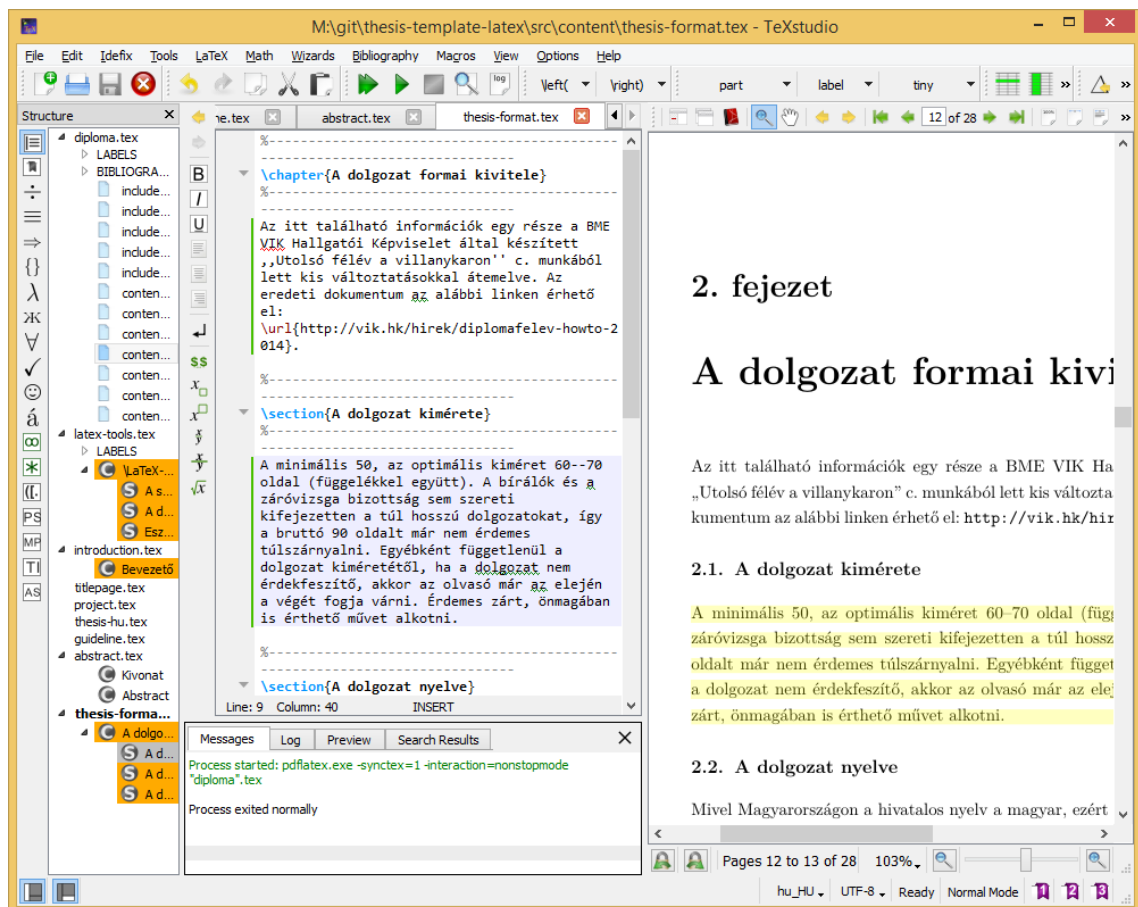
Ez nem kötelező, akár törölhető is. Ha a szerző szükségét érzi, itt lehet köszönetet nyilvánítani azoknak, akik hozzájárultak munkájukkal ahhoz, hogy a hallgató a szakdolgozatban vagy diplomamunkában leírt feladatokat sikeresen elvégezze. A konzulensnek való köszönetnyilvánítás sem kötelező, a konzulensnek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

Irodalomjegyzék

- [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. évf. (1986. 01) 1. sz., 72–76. p. 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, 2000. 04.
- [5] Wai L. Lee–Charles G. Sodini: A topology for higher order interpolative coders. In *Proc. of the IEEE International Symposium on Circuits and Systems* (konferenciaanyag). 1987. 4-7 05., 459–462. p.
- [6] Alexey Mkrtychev: Models for the logic of proofs. In Sergei Adian–Anil Nerode (szerk.): *Logical Foundations of Computer Science*. Lecture Notes in Computer Science sorozat, 1234. köt. 1997, Springer Berlin Heidelberg, 266–275. p. ISBN 978-3-540-63045-6. URL http://dx.doi.org/10.1007/3-540-63045-7_27.
- [7] Richard Schreier: *The Delta-Sigma Toolbox v5.2*. Oregon State University, 2000. 01. <http://www.mathworks.com/matlabcentral/fileexchange/>.
- [8] Ferenc Wettl–Gyula Mayer–Péter Szabó: *L^AT_EX kézikönyv*. 2004, Panem Könyvkiadó.

Függelék

F.1. A TeXstudio felülete



F.1.1. ábra. A TeXstudio \LaTeX -szerkesztő.

F.2. Válasz az „Élet, a világmindenség, meg minden” kérdésre

A Pitagorasz-tételből levezetve

$$c^2 = a^2 + b^2 = 42. \quad (\text{F.2.1})$$

A Faraday-indukciós törvényből levezetve

$$\text{rot } E = -\frac{dB}{dt} \quad \longrightarrow \quad U_i = \oint_{\mathbf{L}} \mathbf{E} d\mathbf{l} = -\frac{d}{dt} \int_A \mathbf{B} d\mathbf{a} = 42. \quad (\text{F.2.2})$$