



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMATIZACE A INFORMATIKY

INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

REALIZACE ŠABLONY ZÁVĚREČNÝCH PRACÍ V LATEX

IMPLEMENTATION OF THE THESIS TEMPLATE IN LATEX

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Antonín Sochor

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Kamil Staněk

BRNO 2023

Zadání bakalářské práce

Ústav: Ústav automatizace a informatiky
Student: **Antonín Sochor**
Studijní program: Strojírenství
Studijní obor: Základy strojního inženýrství
Vedoucí práce: **Ing. Kamil Staněk**
Akademický rok: 2022/23

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Realizace šablony závěrečných prací v LATEX

Stručná charakteristika problematiky úkolu:

Stále větší obliba využití systému LATEX pro velmi kvalitní sazbu dokumentů vyžaduje i dobrou přípravu. S využitím šablon, může pak mnohem více lidí používat tento vysoce pokročilý systém sazby. Připravit šablonu pro bakalářské, diplomové a disertační práce je zcela jistě rozumný krok pro budoucnost. Výsledný produkt univerzální šablony pro využití na VUT FSI si najde jistě mnoho zájemců. Návrh a realizace této univerzální šablony je tématem práce.

Cíle bakalářské práce:

- Analýza stávajících editorů LATEX.
- Zvolení vhodného LATEX editoru pro široké využití.
- Vytvoření šablony bakalářské, diplomové a disertační práce pro VUT FSI.
- V práci mít popis používaných LATEX příkazů s příklady použití.
- V kódu používat podrobné a srozumitelné české popisy.
- Vytvořená šablona musí splňovat stávající požadavky závěrečných prací.
- Podrobná dokumentace řešení.

Seznam doporučené literatury:

LAMPORT, Leslie a Duane BIBBY. LATEX: a document preparation system: user's guide and reference manual. 2nd ed. Boston: Addison-Wesley Pub., 1994. ISBN 978-0201529838.

RYBIČKA, Jiří. LATEX pro začátečníky. 3. vyd. Brno: Konvoj, 2003. ISBN 80-7302-049-1.

KOPKA, Helmut a Patrick W. DALY. A Guide to LATEX. 4rd ed. Boston: Addison-Wesley, c2004. ISBN 978-0321173850.

MITTELBACH, Frank.I a Michel GOOSSENS. The Latex Companion. 2nd ed. Boston: Addison-Wesley, 2004. ISBN 978-0201362992.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2022/23

V Brně, dne

L. S.

doc. Ing. Radomil Matoušek, Ph.D.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

ABSTRAKT

L^AT_EX jako nástroj pro profesionální sazbu dokumentů je velmi oblíbený a jeho obliba stále roste. Nepřeberné množství balíčků, které rozšiřují jeho základní funkcionality, umožňuje L^AT_EX využít v nejrůznějších odvětvích. Mnohé univerzity mají vytvořenou svou vlastní šablonu pro tvorbu závěrečných prací v systému L^AT_EX, a proto vznikla myšlenka vytvořit na pochopení jednoduchou a uživatelsky přívětivou šablonu i pro Fakultu strojního inženýrství na Vysokém učení technickém v Brně. Kromě přehledu online i offline/desktopových L^AT_EX editorů se tato bakalářská práce věnuje popisu fungování vytvořené šablony včetně množství příkladů a poslouží tak jako příručka pro budoucí studenty závěrečných ročníků, kteří se rozhodnou využít L^AT_EX pro svou závěrečnou práci.

ABSTRACT

L^AT_EX as a professional document typesetting tool, is widespread, and its popularity is growing. Many packages extending its basic functionality allow L^AT_EX to be used in various industries. Many universities use the L^AT_EX to create templates for final theses. Therefore the idea of creating an easy-to-understand template for the Faculty of Mechanical Engineering at Brno University of Technology was born. In addition to an overview of online and offline/desktop L^AT_EX editors, this bachelor's thesis is dedicated to being a user-friendly manual of the created template, including several examples. It thus serves as a guide for future final-year students who decide to use L^AT_EX for their final thesis.

KLÍČOVÁ SLOVA

L^AT_EX šablona, šablona závěrečných prací, programování v L^AT_EX, T_EX, L^AT_EX editor, formátování textu v L^AT_EX

KEYWORDS

L^AT_EX template, final thesis template, L^AT_EX programming, T_EX, L^AT_EX editor, text formatting in L^AT_EX

BIBLIOGRAFICKÁ CITACE

SOCHOR, Antonín. *Realizace šablony závěrečných prací v LATEX*. Brno, 2023. Dostupné také z: <https://www.vut.cz/studenti/zav-prace/detail/145841>. Bakalářská práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automatizace a informatiky. Vedoucí práce: Ing. Kamil Staněk.

ČESTNÉ PROHLÁŠENÍ

Prohlašuji, že tato práce je mým původním dílem, vypracoval jsem ji samostatně pod vedením vedoucího práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury.

Jako autor uvedené práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních, a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků.

V Brně dne 26. 5. 2023

.....

Antonín Sochor

PODĚKOVÁNÍ

Rád bych poděkoval svému vedoucímu práce Ing. Kamilu Staňkovi za vedení, pravidelné konzultace, věcné návrhy a připomínky a hodnocení jednotlivých etap tvorby a psaní bakalářské práce. Zároveň děkuji své rodině a kamarádům za podporu nejen při psaní této závěrečné práce.

OBSAH

ÚVOD	15
1 PŘEHLED SOUČASNÉHO STAVU POZNÁNÍ	17
1.1 Systémy T _E X a L ^A T _E X	17
1.2 Přehled L ^A T _E X editorů	18
1.2.1 T _E Xmaker	18
1.2.2 T _E Xstudio	19
1.2.3 LyX	20
1.2.4 Overleaf	21
1.2.5 Papeeria	22
2 VLASTNÍ ŘEŠENÍ	25
2.1 Volba vhodného L ^A T _E X editoru	25
2.2 Základní rozdělení šablony	26
2.3 Uživatelské nastavení	26
2.3.1 Základní údaje o závěrečné práci	26
2.3.2 Vzhled dokumentu	29
2.4 Hlavní pracovní soubor	30
2.4.1 Základní nastavení dokumentu	30
2.4.2 Vlastní text práce	31
2.4.3 Seznamy	32
2.5 Fungování šablony	34
2.5.1 Definice metadat šablony	34
2.5.2 Definice základních příkazů a dat	35
2.5.3 Úvodní textové náležitosti	40
2.5.4 Formátování textu a vzhled stránky	44
2.5.5 Úpravy dalších částí šablony	50
2.5.6 Ostatní pomocné příkazy	52
2.5.7 Multijazykové textové definice	53
3 ZHODNOCENÍ A DISKUZE	55
ZÁVĚR	57
LITERATURA	60
SEZNAM OBRÁZKŮ	61
A Příloha č. 1 – soubor ČTI MĚ	63

ÚVOD

L^AT_EX je nejvýznamnější a nejrozšířenější nadstavba systému T_EX. Byla vytvořena Leslie Lamportem za účelem zpřístupnění tohoto systému co nejvíce uživatelům. Díky tomu tak není nutné přemýšlet, jak tvořený text naformátovat a vysázet, ale pouze jej sepsat a o překlad zdrojového textu a vysázení dokumentu se postará L^AT_EX. O vysoké popularitě tohoto systému svědčí stále se rozšiřující množství balíčků, které přidávají desítky funkcionalit a vylepšení dle potřeb a požadavků uživatelů. Velkou výhodou systému L^AT_EX je jeho univerzálnost. Jádrem celého systému je překladač jazyka T_EX, který je na všech platformách totožný. Uživatel tak má možnost využít libovolný operační systém i editor.

Jeden z rozšiřujících balíčků je i předmětem této bakalářské práce – šablona závěrečných prací pro Fakultu strojního inženýrství na Vysokém učení technickém v Brně. Následující řádky tak slouží i jako referenční příručka k užívání této šablony. Uživatel se dozví, co všechno šablona obsahuje a jak vypadají a fungují její jednotlivé části. Zároveň nalezne seznam nejpoužívanějších L^AT_EX editorů se souhrnem jejich výhod a nevýhod a také výběr vhodných editorů pro široké využití.

Nedílnou součástí závěrečné práce je příloha ve formě PDF dokumentu „Čti mě“, který obsahuje popis vstupního nastavení šablony, tj. kde vyplnit informace o autorovi, o závěrečné práci, jak vložit úvodní stranu, zadání závěrečné práce a další informace, které doplňují vlastní textový obsah práce.

Šablona je dostupná ke stažení na serveru GitHub: <https://bit.ly/FSI-sablona>.

1 PŘEHLED SOUČASNÉHO STAVU POZNÁNÍ

Tato kapitola se v úvodní části věnuje krátkému představení systému $\text{T}_{\text{E}}\text{X}$ a jeho nadstavbě $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$, v další části je poté sestaven přehled nejpoužívanějších $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ editorů, a to jak online, tak i offline/desktopových aplikací.

1.1 Systémy $\text{T}_{\text{E}}\text{X}$ a $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$

Pro formátování vstupního textu slouží formátovací programy. Mezi ně se řadí i $\text{T}_{\text{E}}\text{X}$, který byl vytvořen Donaldem E. Knuthem. Účelem bylo vytvořit program pro vysoce profesionální a kvalitní sazbu dokumentů z matematického a technického prostředí. $\text{T}_{\text{E}}\text{X}$ však umožňuje pouze primitivní operace pro sázení dokumentů, které jsou řízeny sadou jednoduchých příkazů, tzv. primitivů. [1]

Primitivy umožňují definovat tzv. makra, tedy složitější a komplexnější příkazy. Tato možnost byla využita Leslie Lamportem, který naprogramoval sadu maker. Vzniklá sada dostala název $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ a dnes je nejrozšířenější nadstavbou základního jazyka $\text{T}_{\text{E}}\text{X}$. Lamportem naprogramovaná makra umožňují členit dokumenty na kapitoly, přidávat citace, využívat výhod automatického číslování rovnic, používat křížové odkazy nebo jedním příkazem generovat obsah celého dokumentu. Vznikl tak nástroj umožňující užívání celého systému i bez znalostí programování. [1]

Pomocí vytvořených příkazů uživatelé zadávají, co chtějí vysázet. Informace o tom, jak se má zadaný textový řetězec vysázet (velikost písma, odsazení, umístění na straně a další), jsou obsaženy v použitém příkazu. Uživatel díky tomu nepotřebuje typografické znalosti, protože systém $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ za něj všechny vstupní textové řetězce vysází sám. [2]

Systém $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ rovněž umožňuje importování balíčků (sad dalších maker), které rozšiřují základní funkcionality o další prvky. Díky tomu se celý systém rozšířil i do oblastí, pro které nebyl zamýšlen. [1]



Obrázek 1: Logo systému $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$. [vlastní]

$\text{T}_{\text{E}}\text{X}$ a $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ jsou rovněž nazývány programovacími jazyky. Jejich užití není limitováno žádnými jinými programy, aplikacemi nebo operačními systémy. Pro překlad vstupního textového dokumentu uživatel potřebuje pouze nainstalovanou $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ distribuci a libovolný program s překladačem jazyka $\text{T}_{\text{E}}\text{X}$. [1]

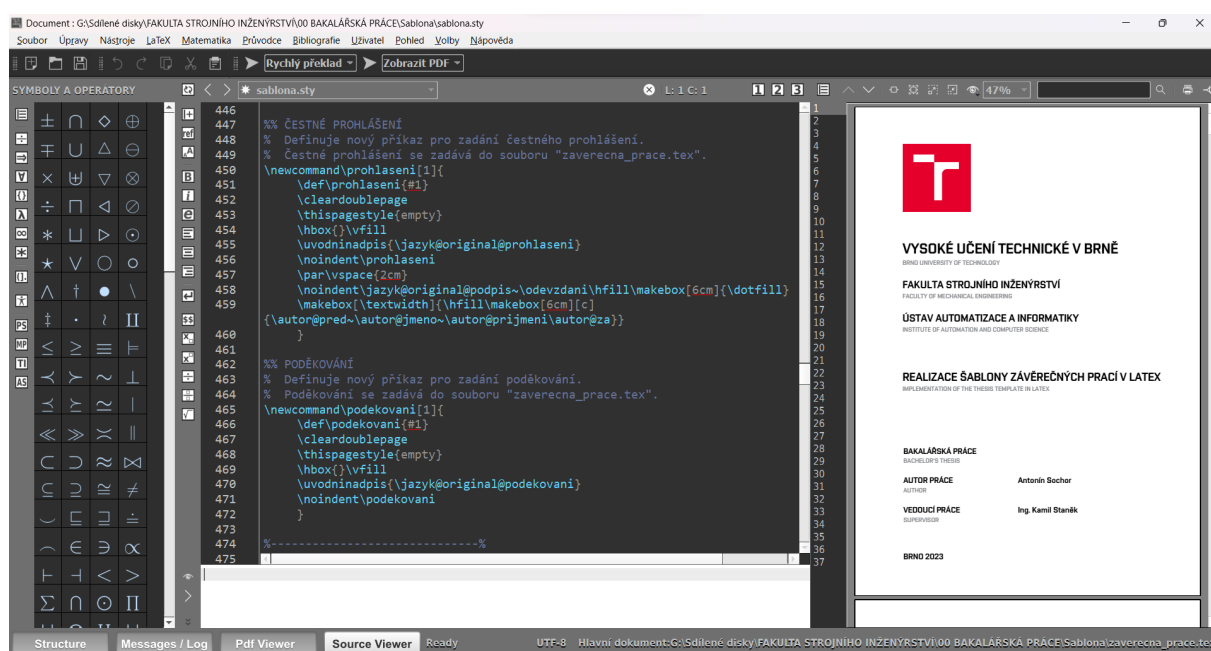
1.2 Přehled L^AT_EX editorů

Pro tvorbu L^AT_EX dokumentů existují editory, které uživatelům usnadňují práci. Tyto editory nabízejí barevné rozlišení příkazů pro snadnější orientaci v textu, našeptávání jednotlivých příkazů, možnost úpravy vizuálního vzhledu editoru a další funkcionality. Společným základem všech editorů je integrovaný překladač jazyka T_EX, který ze zdrojového kódu (textový soubor s příkazy a vlastním textovým obsahem) vytvoří požadovaný dokument, například ve formátu PDF. L^AT_EX dokument lze napsat také v obyčejném poznámkovém bloku, uživatel v něm ale nemá možnost text dále jakkoliv efektivně zpracovat.

Na následujících řádcích byl vytvořen přehled nejpoužívanějších L^AT_EX editorů s jejich popisem, ukázkou a přehledem výhod a nevýhod. Tento přehled vychází z několika webových příspěvků věnujících se sestavením žebříčku L^AT_EX editorů [3] [4] [5].

1.2.1 T_EXmaker

T_EXmaker editor vznikl v roce 2003 a dnes se řadí k nejoblíbenějším a nejpoužívanějším L^AT_EX desktopovým editorům. Nabízí přehledné prostředí s podporou českého jazyka, integrovaný PDF prohlížeč tvořeného dokumentu a dnes velmi oblíbený dark-mode uživatelského prostředí. Český jazyk je integrován i na úrovni kontroly pravopisu v reálném čase, tj. během psaní textu. [6]



Obrázek 2: Ukáзка prostředí T_EXmaker editoru. [vlastní]

Autoři poskytují řešení pro operační systémy Windows, macOS a Linux. Pro 370 nejpoužívanějších matematických symbolů má T_EXmaker samostatná tlačítka, díky kterým není nutné znát jednotlivé příkazy na požadované matematické symboly. Při výskytu chyb editor uživatele upozorní a zobrazí čísla řádků, na kterých se daná chyba nachází. Velkou

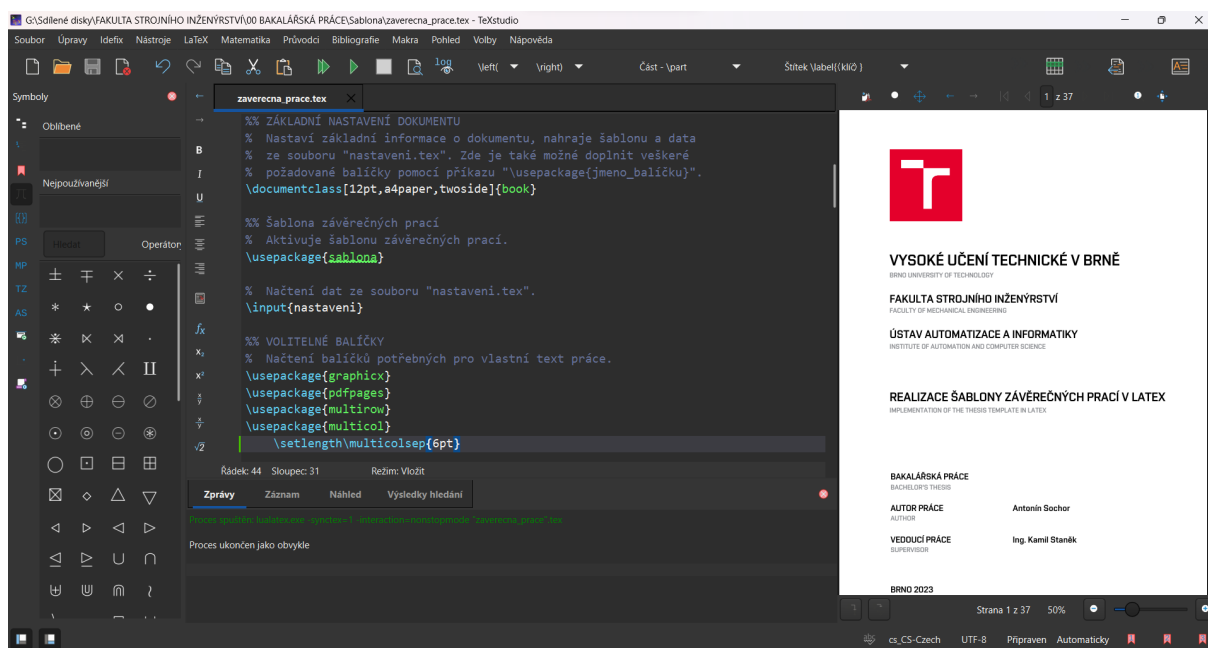
výhodou je našeptávání základních \LaTeX příkazů. Do editoru je integrováno několik překladačů, mezi nimiž může uživatel volně přepínat. Na výběr je \LaTeX , pdf \LaTeX , $\text{\LaTeX}mk$, $\text{Xe}\text{\LaTeX}$ a $\text{Lua}\text{\LaTeX}$. [6]

Výhody a nevýhody:

- + použití zdarma
- + uživatelské prostředí v českém jazyce
- + kontrola pravopisu
- + verze pro Windows, macOS i Linux
- + tlačítka pro matematické symboly
- + našeptávání příkazů
- + volba překladače zdrojového kódu
- + integrovaný PDF prohlížeč
- chybějící překlad některých tlačítek
- nevyužívá cloud

1.2.2 \TeX studio

Cílem \TeX studio editoru je udělat \LaTeX tvorbu co nejjednodušší a nejpohodlnější [7].



Obrázek 3: Ukázka prostředí \TeX studio editoru. [vlastní]

\TeX studio editor je dostupný na platformách Windows, Linux a macOS. Nabízí české uživatelské prostředí s kontrolou pravopisu, našeptávání \LaTeX příkazů, asistenta pro vkládání obrázků včetně podpory Drag & Drop a asistenta při tvorbě tabulek. Editor má také integrovaný PDF prohlížeč překládaného dokumentu. Velkou výhodou je víceprůchodový překlad, při kterém editor automaticky spustí potřebné překladače a uživatel je tak nemusí spouštět manuálně. Editor podporuje uživatelskou volbu překladačů. Na výběr jsou \LaTeX , pdf \LaTeX , $\text{\LaTeX}mk$, $\text{Xe}\text{\LaTeX}$ a $\text{Lua}\text{\LaTeX}$. [8]

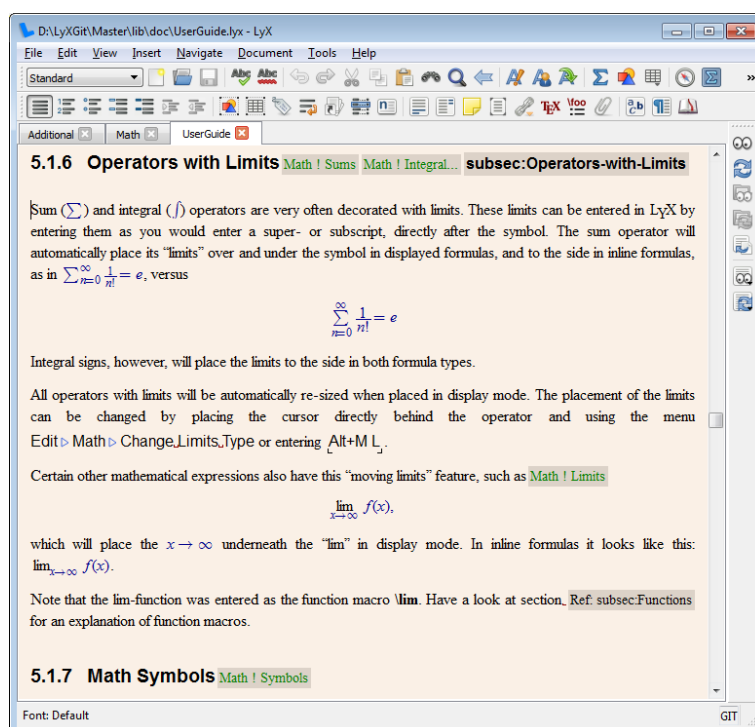
Nevýhodou je ve výchozím stavu nepřehledně nastavené zvýrazňování \LaTeX příkazů.

Výhody a nevýhody:

- | | |
|--|---|
| <ul style="list-style-type: none"> + použití zdarma + uživatelské prostředí v českém jazyce + kontrola pravopisu + verze pro Windows, macOS i Linux + tlačítka pro matematické symboly + našeptávání příkazů + integrovaný PDF prohlížeč + automatický víceprůchodový překlad + asistent vkládání obrázků + asistent tvorby tabulek + podpora Drag & Drop | <ul style="list-style-type: none"> – chybějící překlad některých tlačítek – nevyužívá cloud – nepřehledné výchozí zvýrazňování |
|--|---|

1.2.3 LyX

Editor LyX je výkonný WYSIWYM editor. WYSIWYM je zkratkou anglické věty „What You See Is What You Mean“, v češtině „Vidíš to, co máš na mysli“. Jedná se tedy o editor, ve kterém uživatel zpracovává dokument podobně jako například v aplikaci Microsoft Word. Píše text, který člení na kapitoly, pomocí integrovaných nástrojů vkládá obrázky, rovnice a tabulky. Uživatel tak nevyužívá \LaTeX příkazy, tvoří pouze vlastní obsah. O doplnění příkazů a překlad dokumentu pomocí jazyka \TeX se stará editor. [9]



Obrázek 4: Ukázka prostředí LyX editoru. [10]

LyX je možné spustit na operačních systémech Windows, Linux i macOS. Podporuje ukládání a správu jednotlivých verzí, uživatelské prostředí je přeloženo do českého jazyka

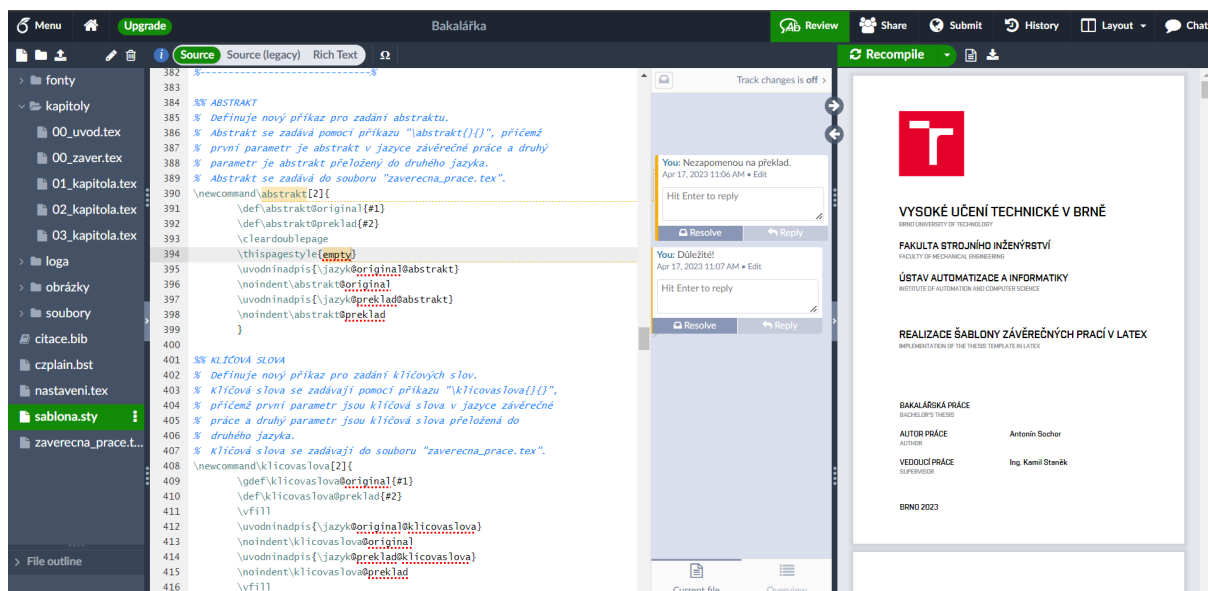
včetně podpory kontroly pravopisu. Editor má zabudovaný vlastní PDF prohlížeč tvořeného dokumentu. Autoři LyX editoru uživatelům umožňují volbu překladače. Na výběr jsou \LaTeX , pdf \LaTeX , Xe \LaTeX a Lua \LaTeX . [11]

Výhody a nevýhody:

- + použití zdarma
- + uživatelské prostředí v českém jazyce
- + kontrola pravopisu
- + verze pro Windows, macOS i Linux
- + integrovaný PDF prohlížeč
- + WYSIWYM editor
- + psaní bez \LaTeX příkazů
- pracnější tvorba matematických rovnic
- nevyužívá cloud

1.2.4 Overleaf

Overleaf je moderní online \LaTeX editor, který běží na serverech poskytovatele, a uživatel tak není nucen si jej instalovat přímo do svého zařízení. K editoru se tak dostane přes libovolný webový prohlížeč. Editor se vyznačuje svou jednoduchostí a rychlostí. Nabízí registraci a používání zdarma s možností předplacení prémiové verze. Overleaf je částečně přeložen do českého jazyka, tento překlad je ale nedokonalý a neúplný – v uživatelském prostředí jsou přeloženy pouze některé položky. Velkou výhodou je automatický víceprůchodový překlad. Overleaf také nabízí výběr z více překladačů. Integrovány jsou \LaTeX , pdf \LaTeX , Xe \LaTeX a Lua \LaTeX . [12]



Obrázek 5: Ukázka prostředí Overleaf editoru. Upraveno. [vlastní]

Neplacená verze editoru nabízí veškeré základní funkce včetně volby překladače zdrojového kódu a možnost spolupráce s jedním dalším uživatelem. Jediným omezením je časový limit pro překlad. Překladač jazyka \TeX má na přeložení zdrojového kódu a vygenerování výsledného dokumentu pouze jednu minutu. [13]

Prémiová verze je rozdělena na tři balíčky, které se liší svou cenou a obsahem prémiových funkcí. K těm patří možnost spolupráce s více osobami současně, ukládání jednotlivých verzí dokumentu s možností jejich obnovení, integrace GitHubu, Dropboxu a citačních manažerů Mendeley a Zotero. Overleaf nabízí také levnější verzi předplatného pro studenty. [14]

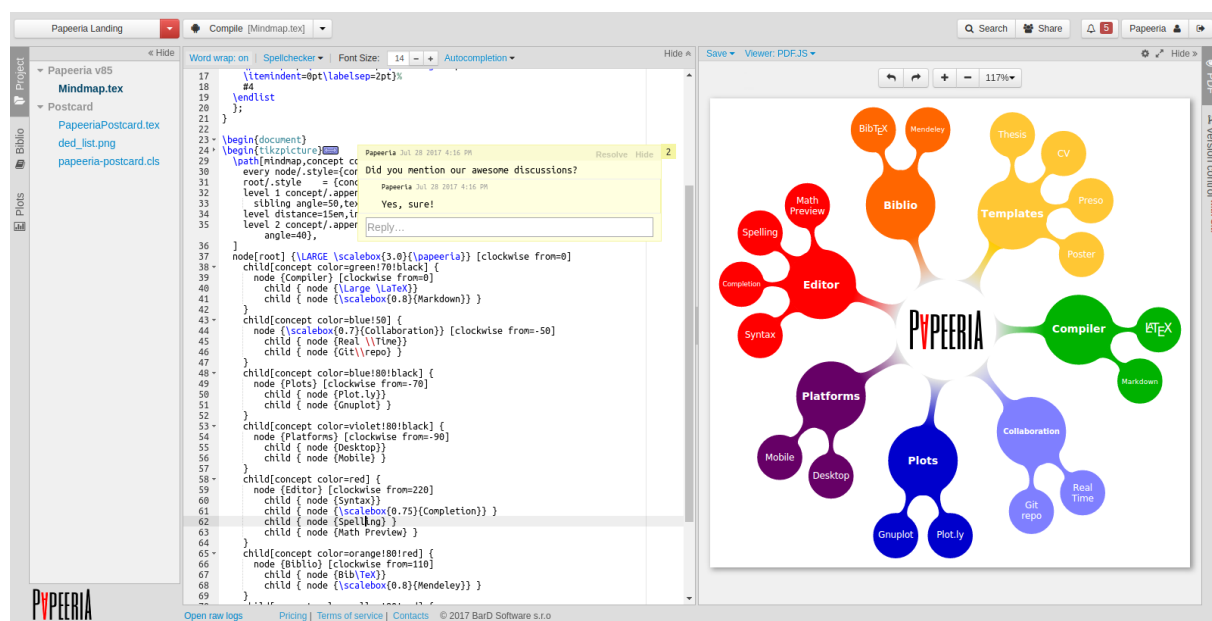
Overleaf má nastaven limit pro počet souborů v rámci jednoho projektu. Pro každý projekt může uživatel nahrát maximálně 2000 souborů. Tento limit tak může být komplikací pro větší a náročnější projekty. [15]

Výhody a nevýhody:

- | | |
|--|---|
| <ul style="list-style-type: none"> + použití zdarma (s omezením) + neomezené množství projektů + našeptávání příkazů + možnost spolupráce s dalšími uživateli + integrovaný PDF prohlížeč | <ul style="list-style-type: none"> – nedokonalý překlad do českého jazyka – vysoké ceny prémiových funkcí – omezený čas pro překlad v základní verzi |
|--|---|

1.2.5 Papeeria

Papeeria je online \LaTeX editor nabízející placenou verzi i verzi zdarma. Neplacená verze nabízí neomezené množství projektů s možností spolupráce s jinými uživateli, Git synchronizaci a možnost ukládání historie změn jednotlivých dokumentů po dobu 24 hodin. Veškeré projekty jsou veřejné, uživatel má ve verzi zdarma možnost pouze jednoho soukromého projektu. [16] [17]



Obrázek 6: Ukázka prostředí Papeeria editoru. [18]

Placená verze přidává možnost synchronizace s DropBoxem a Google Drivem, až deset soukromých projektů, prioritní kompilace zdrojového kódu na serveru. Uživatel má mož-

nost obnovit historii změn dokumentů za posledních 30 dnů a také možnost využít citační manažer Mendeley. [17]

Editor uživateli nabízí možnost volby překladače jazyka $\text{T}_{\text{E}}\text{X}$ ($\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, pdf $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, $\text{XeL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ a $\text{LuaL}^{\text{A}}\text{T}_{\text{E}}\text{X}$) a zobrazit výsledný dokument v integrovaném PDF prohlížeči. [16]

Velkou nevýhodou Papeeria editoru je jeho zastaralost. Poslední aktualizace je z roku 2020. Nejnovější verze distribuce jazyka $\text{T}_{\text{E}}\text{X}$, kterou má uživatel možnost nastavit, je z roku 2019. Editor nenabízí českou lokaci. [16]

Výhody a nevýhody:

- | | |
|--|--|
| + použití zdarma (s omezením) | – pouze jeden soukromý projekt. |
| + neomezené množství projektů | – absence překladu prostředí do českého jazyka |
| + našeptávání příkazů | – zastaralá verze jazyka $\text{T}_{\text{E}}\text{X}$ |
| + možnost spolupráce s dalšími uživateli | – poslední aktualizace v roce 2020 |
| + integrovaný PDF prohlížeč | – problémové nahrávání souborů |
| + možnost předplatného | |
| + ukládání historie změn | |

2 VLASTNÍ ŘEŠENÍ

Snahou autora bylo vytvořit šablonu tak, aby s ní dokázal pracovat i úplný laik. Z tohoto důvodu je i tato bakalářská práce psána záměrně jednoduchým stylem, aby bylo možné ji použít i jako referenční příručku pro práci se šablonou.

Na několika fakultách VUT jsou dostupné jejich vlastní L^AT_EX šablony, které jsou však v některých případech zbytečně složité. Jejich zdrojový kód obsahuje několik tisíc řádků, z nichž stovky jsou ve formě poznámek vzniklých jednotlivými úpravami a zásahy do původního kódu. Autorem vytvořená šablona je strukturována do souvisejících celků, které jsou ve zdrojovém kódu přehledně odděleny. Jednotlivé části těchto celků jsou popsány tak, aby je bylo možné lépe pochopit a používat v praxi.

2.1 Volba vhodného L^AT_EX editoru

Jedním z cílů této bakalářské práce je volba vhodného L^AT_EX editoru pro široké využití. Tato volba vychází z rešerše L^AT_EX editorů v kapitole 1.2.

Na základě vlastních zkušeností autora této závěrečné práce je doporučován offline/desktopový editor T_EXmaker a online editor Overleaf.

T_EXmaker

Editor, ve kterém vznikla celá tato závěrečná práce i samotná šablona. Autor oceňuje přehledné a jednoduché nastavení editoru s možností programování tlačítka „*Rychlý překlad*“ pro nejrychlejší a nejpohodlnější způsob překladu zdrojového textu. Velkou výhodou autor nachází také v možnosti rychlého přepínání zabudovaných překladačů jazyka T_EX.

Velkým pomocníkem při tvorbě práce byla možnost zvolení hlavního dokumentu. Editor umožňuje při otevření více souborů současně volbu „*Prohlásit současný dokument za hlavní*“. To znamená, že autor práce nemusel myslet na to, ve kterém souboru se zrovna nachází, ale mohl si nastavit jeden soubor jako hlavní, který se bude automaticky překládat. V případě této závěrečné práce se vždy jednalo o hlavní pracovní soubor `zaverecna_prace.tex`.

Autorem práce nalezená nevýhoda je nemožnost vytvářet poznámky či značky v jednotlivých dokumentech a označit si tak místo, ke kterému se potřebuje později vrátit, případně místo, které je problémové. Další nevýhodou je pomalejší překlad oproti Overleaf editoru.

Overleaf

Editor Overleaf byl využíván souběžně s editorem T_EXmaker pro kontrolu funkčnosti šablony i v online prostředí. Za neocenitelnou funkcionalitu tohoto editoru je autorem práce považováno automatické přepínání překladačů jazyka T_EX a víceprůchodový překlad. Není potřeba přepínat mezi překladačem jazyka T_EX a programem BIBT_EX pro překlad citací, Overleaf udělá vše automaticky a dokument zároveň přeloží víceprůchodově, takže je hned vidět výsledný dokument ve finální verzi.

Zmiňovaná nevýhoda \TeX maker editoru v podobně vytváření poznámek se zde nenachází. Overleaf nabízí možnost komentování jednotlivých příkazů zdrojového textu a také zobrazení seznamu těchto komentářů. Editor Overleaf běží na serverech poskytovatele, které jsou (pravděpodobně) výkonnější než počítačová sestava autora práce, proto editor nabízí rychlejší překlad zdrojového kódu dokumentu.

2.2 Základní rozdělení šablony

Šablona závěrečných prací pro svou funkčnost vyžaduje tři soubory, které dohromady utvářejí celou šablonu:

1. `sablona.sty`,
2. `zaverecna_prace.tex`,
3. `nastaveni.tex`.

V adresáři se šablonou je také vytvořeno několik složek, ve kterých jsou uloženy další soubory, které jsou v šabloně využívány. Jedná se o složky:

1. `fonty` – obsahuje soubory VUT fontů, které šablona využívá pro tvorbu hlavičky každé strany dokumentu;
2. `kapitoly` – složka pro vkládání souborů s textovým obsahem jednotlivých kapitol;
3. `loga` – obsahuje loga VUT a FSI, která šablona vkládá do hlavičky dokumentu;
4. `obrazky` – složka pro ukládání obrázků použitých v závěrečné práci;
5. `soubory` – obsahuje PDF soubory úvodní strany a zadání závěrečné práce stažené ze Studisu.

Funkce, nastavení a použití těchto souborů jsou popsány v následujících podkapitolách.

2.3 Uživatelské nastavení

Uživatel provádí dostupná nastavení šablony v souboru `nastaveni.tex`. Jedná se o textový soubor obsahující několik vytvořených příkazů. Každý příkaz má jeden či více argumentů, které může uživatel libovolně měnit. Tyto příkazy jsou rozděleny do dvou ucelených částí:

1. základní údaje o závěrečné práci,
2. vzhled dokumentu.

2.3.1 Základní údaje o závěrečné práci

V této části uživatel vyplňuje údaje o druhu závěrečné práce (bakalářská, diplomová, dizertační), název samotné práce, jméno autora a vedoucího, datum odevzdání a další.

Druh závěrečné práce

Příkaz: `\prace{}`.

Výchozí nastavení: `\prace{bc}`.

Jako argument slouží zkratky:

- `bc` = bakalářská práce,
- `ing` = diplomová práce,
- `phd` = dizertační práce.

Jednotlivé možnosti jsou předvyplněny – uživatel tak pouze vybere požadovaný typ závěrečné práce. Zbylé možnosti je možné smazat anebo okomentovat pomocí symbolu `%` na začátku řádku.

Pokud uživatel zadá chybnou nebo neznámou zkratku, bude argument nahrazen chybovou hláškou „*Nezadán/neznámý typ práce*“.

Název závěrečné práce

Příkaz: `\nazev{}`.

Výchozí nastavení: bez argumentu.

Příklad: `\nazev{Realizace šablony závěrečných prací v LATEX}`.

Název závěrečné práce se zadává celý. Je možné jej zkopírovat ze Studisu nebo ze zadání práce.

Odkaz na elektronickou publikaci závěrečné práce

Příkaz: `\odkaz{}`.

Výchozí nastavení: bez argumentu.

Příklad: `\odkaz{https://www.vut.cz/studenti/zav-prace/detail/145841}`.

Odkaz na elektronickou publikaci je vygenerován ve Studisu a do práce se vkládá celý.

Autor závěrečné práce

Příkaz: `\autor{}{}{}{}`.

Výchozí nastavení: bez argumentů.

Příklad: `\autor{Ing.}{Student}{Zkoušený}{Ph.D.}`.

Příkaz má čtyři argumenty v následujícím pořadí:

- tituly před jménem,
- jméno,
- příjmení,
- tituly za jménem.

Pokud má autor práce více titulů před jménem, odděluje je nezlomitelnou mezerou pomocí symbolu `~`, pokud má více titulů za jménem, odděluje je čárkou a nezlomitelnou mezerou pomocí symbolů `,~`.

Příklad: `\autor{prof.~JUDr.~Ing.}{Jan}{Novák}{Ph.D.,~dr. h. c.}`.

V případě, že autor tituly nemá, ponechají se příslušné argumenty prázdné.

Příklad: `\autor{}{Student}{Zkoušený}{}`.

Vedoucí závěrečné práce

Příkaz: `\vedoucí{ }{ }{ }`.

Výchozí nastavení: bez argumentů.

Příklad: `\vedoucí{Ing.}{Vedoucí}{Rozumný}{Ph.D.}`.

Pro zadání vedoucího práce platí stejná pravidla a podmínky jako pro autora popisovaného výše.

Ústav

Příkaz: `\ustav{ }`.

Výchozí nastavení: `\ustav{Ústav automatizace a informatiky}`.

Jako argument příkazu se zadává název ústavu, na kterém autor studuje, případně ten, na kterém dělá a bude obhajovat svou závěrečnou práci (studenti z obecného bakalářského oboru). Všechny fakultní ústavy jsou vyplněny, uživatel tak pouze daný ústav vybere. Zbylé ústavy je možné smazat anebo zakomentovat pomocí symbolu %.

Na výběr jsou následující ústavy:

1. Ústav matematiky,
2. Ústav fyzikálního inženýrství,
3. Ústav mechaniky těles, mechatroniky a biomechaniky,
4. Ústav materiálových věd a inženýrství,
5. Ústav konstruování,
6. Energetický ústav,
7. Ústav strojírenské technologie,
8. Ústav výrobních strojů, systémů a robotiky,
9. Ústav procesního inženýrství,
10. Ústav automobilního a dopravního inženýrství,
11. Letecký ústav,
12. Ústav automatizace a informatiky,
13. Ústav jazyků.

Město

Příkaz: `\mesto{ }`.

Výchozí nastavení: `\mesto{Brno}`.

Příkaz nastavuje město, ve kterém bude probíhat SZZ a obhajoba závěrečné práce. Typicky se jedná o Brno.

Rok obhajoby

Příkaz: `\obhajoba{ }`.

Výchozí nastavení: bez argumentu.

Příklad: `\obhajoba{2023}`.

Argumentem tohoto příkazu je rok, ve kterém proběhne SZZ a obhajoba práce. Nejedná se o akademický rok.

Argument se zadává ve formátu RRRR (R = rok).

Datum odevzdání

Příkaz: `\odevzdani{}`.

Výchozí nastavení: `bez argumentu`.

Příklad: `\odevzdani{26. 05. 2023}`.

Příkaz nastavuje datum odevzdání závěrečné práce do Studisu. Uživatel uvádí datum, kdy svou závěrečnou práci nahrál do systému Studis, případně termín, do kdy je možné práci odevzdat.

Argument se zadává ve formátu DD. MM. RRRR (D = den, M = měsíc, R = rok).

Jazyk zpracování

Příkaz: `\jazyk{}`.

Výchozí nastavení: `\jazyk{cz-en}`.

Argumentem příkazu je zkratka jazyků (hlavní a vedlejší), ve kterých je závěrečná práce napsána. Hlavní jazyk je jazyk, ve kterém je napsán vlastní textový obsah práce, vedlejší jazyk je použit pro sekundární nadpisy na titulní straně a pro překlad abstraktu a klíčových slov. Uživatel si vybere požadovanou volbu z předvyplněných možností.

Argument je ve formátu `xx-yy` (xx = zkratka hlavního jazyka, yy = zkratka vedlejšího jazyka). Šablona podporuje následující jazykové mutace:

- `cz-en` hlavní jazyk: čeština, vedlejší jazyk: angličtina;
- `sk-en` hlavní jazyk: slovenština, vedlejší jazyk: angličtina;
- `en-cz` hlavní jazyk: angličtina, vedlejší jazyk: čeština;
- `en-sk` hlavní jazyk: angličtina, vedlejší jazyk: slovenština.

Pokud nebude vybrána žádná nebo nesprávná možnost, bude automaticky nastavena volba `cz-en`.

2.3.2 Vzhled dokumentu

Uživatel zde volí styl formátování odstavců a číslování obrázků, tabulek a rovnic.

Formátování odstavců

Příkaz: `\odstavec{}`.

Výchozí nastavení: `\odstavec{2}`.

Jako argument příkazu se zadávají číslíce 1 a 2, které odpovídají následujícím volbám:

1. Nový odstavec bude začínat odstavcovou zarážkou, tj. první slovo nového odstavce bude zaraženo o 1 cm doprava.
2. Nový odstavec bude bez odstavcové zarážky a od předchozího odstavce bude odsazen o 6 pt ve vertikálním směru.

Pokud nebude zvolena žádná možnost anebo možnost nesprávná, bude automaticky použita volba č. 2.

Číslování obrázků, tabulek a rovnic

Příkaz: `\cislovani{}`.

Výchozí nastavení: `\cislovani{2}`.

Příkaz umožňuje nastavit styl číslování. Uživatel má při číslování na výběr ze dvou možností:

1. Číslo obrázku, tabulky a rovnice obsahuje i číslo hlavní kapitoly, tj. číslo bude ve formátu: **Obrázek 1.3: Rozložení pracovních sil ve firmě.**, kde první číslo je číslo hlavní kapitoly a druhé číslo je pořadové číslo obrázku v dané kapitole.
2. Číslo obrázku, tabulky a rovnice neobsahuje číslo kapitoly a bude pouze ve formátu: **Obrázek 3: Rozložení pracovních sil ve firmě.**

Výchozí hodnotou je možnost č. 2, která bude rovněž nastavena v případě nezvolení žádné možnosti nebo vyplnění chybného čísla.

2.4 Hlavní pracovní soubor

Hlavním pracovním souborem je soubor `zaverecna_prace.tex`. Tímto souborem se importují veškerá nastavení a data ze souboru `nastaveni.tex`, načítají se volitelné balíčky, vyplňují se nezbytné textové pasáže (abstrakt, klíčová slova a další), vkládají se jednotlivé kapitoly, generují se jednotlivé seznamy (obrázků, tabulek a rovnic) a obsah celé práce a také se aktivuje samotná šablona závěrečné práce.

Soubor je rozdělen na čtyři celky:

1. základní nastavení dokumentu,
2. vlastní text práce,
3. seznamy,
4. přílohy.

Pozn.: V hlavním pracovním souboru je důležitá posloupnost příkazů. V pořadí, ve kterém jsou příkazy za sebou poskládány, budou systémem \LaTeX také provedeny. Na toto je znovu upozorněno v kritických místech.

2.4.1 Základní nastavení dokumentu

Prvotním nastavením celého dokumentu je volba tzv. třídy dokumentu, velikosti písma a jednostranného či oboustranného tisku. Směrnice děkana č. 3/2022, část F, body 1 a 2 stanovují velikost písma na 12 pt (kromě nadpisů) a použití oboustranného tisku. To je použito jako výchozí hodnota.

Výchozí hodnota: `\documentclass[12pt,a4paper,twoside]{book}`.

Aktivace šablony závěrečných prací a import dat ze souboru `nastaveni.tex` probíhá pomocí příkazů `\usepackage{sablona}` a `\input{nastaveni}`.

Uživatel má možnost ve své práci využívat libovolné \LaTeX balíčky. Jejich užití je limitováno pouze ze strany tvůrců balíčků, pokud ve svých dokumentacích upozorňují na

nekompatibilitu jejich balíčku s jiným, případně nekompatibilitou, která není tvůrci balíčků objevena a ověřena.

V této části hlavního pracovního souboru jsou přednastaveny pouze tři balíčky:

1. **graphicx** balíček pro rozšířenou podporu práce s obrázky a barvami,
2. **pdfpages** balíček umožňující importování stran z jiných PDF dokumentů,
3. **multirow** balíček pro rozšířené možnosti tvorby tabulek.

2.4.2 Vlastní text práce

Nejdůležitější část celého hlavního pracovního souboru. Uživatel zde vkládá úvodní stranu, zadání závěrečné práce, jednotlivé kapitoly, generuje obsah a další.

V pořadí, v jakém budou níže zmíněné příkazy použity, budou také systémem \LaTeX zpracovány a vygenerovány ve výsledném PDF dokumentu.

Titulní strana a zadání

Příkaz: `\includepdf[]{}.`

Výchozí nastavení:

- `\includepdf[pages=-]{soubory/titulni_strana},`
- `\includepdf[pages=-,trim={20mm 0mm 10mm 0mm},offset={5mm 0mm}]{soubory/zadani}.`

Titulní strana a zadání závěrečné práce jsou dokumenty ve formátu PDF. Oba soubory jsou vygenerovány ve Studisu a je možné je stáhnout a uložit do složky **soubory** v adresáři se šablonou. Z důvodu usnadnění práce se šablonou je nutné soubory pojmenovat jako **titulni_strana.pdf** a **zadani.pdf**. Uživatel poté nemusí ručně přepisovat příkaz pro import obou dokumentů a může využít výchozí nastavení. Zadání práce je pomocí argumentů **trim** a **offset** ořezáno a posunuto tak, aby respektovalo okraje stránky o velikosti 3 cm kvůli vazbě dokumentu.

Abstrakt

Příkaz: `\abstrakt{}{}.`

Výchozí nastavení: **bez argumentů**.

Příkaz generující abstrakt. Vyžaduje dva argumenty – první argument je vlastní text abstraktu v hlavním jazyce, druhý argument je poté abstrakt (uživatel) přeložený do vedlejšího jazyka.

Klíčová slova

Příkaz: `\klicovaslova{}{}.`

Výchozí nastavení: **bez argumentů**.

Podobně jako abstrakt se i klíčová slova zadávají v hlavním a vedlejším jazyce. První argument příkazu slouží pro hlavní jazyk, druhý argument pro jazyk vedlejší.

Bibliografická citace

Příkaz: `\citace.`

Dle dat uvedených v souboru `nastaveni.tex` příkaz automaticky vytvoří vlastní bibliografickou citaci autorovi závěrečné práce dle ČSN ISO 690.

Čestné prohlášení

Příkaz: `\prohlaseni{}`.

Výchozí nastavení: `bez argumentu`.

Příkaz sloužící pro zadání prohlášení autora o původnosti práce. Jedná se o poslední úvodní nezbytnou náležitost, kterou musí obsahovat každá závěrečná práce. Uživatel zadává pouze textový obsah. Místo, datum a prostor po vlastnoruční podpis jsou automaticky generovány příkazem.

Poděkování

Příkaz: `\podekovani{}`.

Výchozí nastavení: `bez argumentů`.

Poděkování je nepovinnou částí závěrečné práce. Textový obsah poděkování je vepisován jako argument příkazu.

Obsah

Příkaz: `\obsah`.

Příkaz, který automaticky vygeneruje stranu (či více stran) s obsahem celého dokumentu závěrečné práce.

Kapitoly

Příkaz: `\input{}`.

Závěrečná část celého celku. Uživatel zde vkládá jednotlivé kapitoly. Jedná se o kritické místo na posloupnost vkládání kapitol. V pořadí, v jakém budou kapitoly vloženy, budou také řazeny ve výsledném PDF dokumentu.

Kapitoly je možné psát do jednoho souboru s příponou `.tex` nebo každou kapitolu zvlášť. Soubory s kapitolami je možné ukládat pod libovolnými názvy do složky `kapitoly` v hlavním adresáři šablony.

Kapitoly „Úvod“ a „Závěr“ se sepisují v samostatných souborech s názvy `00_uvod.tex` a `00_zaver.tex`. Tyto soubory jsou pro uživatele vytvořeny ve složce `kapitoly` a vloženy do šablony pomocí příkazů:

- `\input{kapitoly/00_uvod}`,
- `\input{kapitoly/00_zaver}`.

Uživatel tak pouze doplní vlastní textový obsah obou kapitol.

2.4.3 Seznamy

Poslední část hlavního souboru, ve které se generují seznamy použité literatury, obrázků, tabulek a příloh. Každý z příkazů vygeneruje daný seznam a vytvoří k němu patřičný nadpis kapitoly. Tyto kapitoly se ve výchozím stavu neobjevují v obsahu. K zařazení do obsahu slouží příkaz `\addcontentsline{}{}{}`. Aby bylo možné v obsahu kliknout na

libovolný seznam a přesunout se na něj při prohlížení dokumentu, je potřeba vytvořit imaginární kapitolu pomocí příkazu `\phantomsection`. Ten vytvoří bod, do kterého se poté prohlížeč PDF dokumentu přesune.

Literatura

Příkaz: `\bibliography{}`.

Výchozí nastavení:

- `\bibliography{citace}`,
- `\phantomsection`,
- `\addcontentsline{toc}{chapter}{LITERATURA}`.

Příkaz vytvoří seznam použité literatury, registruje kapitolu do obsahu a vytvoří bod, do kterého se bude možné odkázat z obsahu.

Pro tvorbu citací je využíván systém $\text{BIB}\text{T}_{\text{E}}\text{X}$. Informace o citovaném díle se zadávají do souboru `citace.bib`. Citační manažery jako Mendeley, Zotero nebo Citace PRO umožňují jednoduché tvoření citací a jejich následné stažení přímo ve formátu $\text{BIB}\text{T}_{\text{E}}\text{X}$.

Forma citací je dána normou ČSN ISO 690. Na Fakultě informačních technologií VUT vznikl v rámci tvorby jejich šablony citační stylový balíček `czplain.bst`. Tento stylový balíček je využíván i v této šabloně. Jeho zdrojový soubor je uložen v hlavním adresáři šablony. Balíček byl autorem této šablony částečně modifikován, aby respektoval změny v citační normě platné od 1. 12. 2022.

Seznam obrázků

Příkaz: `\listoffigures`.

Výchozí nastavení:

- `\listoffigures`,
- `\phantomsection`,
- `\addcontentsline{toc}{chapter}{\MakeUppercase\listfigurename}`.

Příkaz vytvoří seznam obrázků, registruje jej do obsahu a vytvoří na něj odkaz, na který je v obsahu možné kliknout.

Seznam tabulek

Příkaz: `\listoftables`.

Výchozí nastavení:

- `\listoffigures`,
- `\phantomsection`,
- `\addcontentsline{toc}{chapter}{\MakeUppercase\listtablename}`.

Příkaz vygeneruje seznam tabulek, vloží jeho nadpis do obsahu a vytvoří imaginární bod, do kterého bude obsah odkazovat po kliknutí.

Další seznamy

V případě potřeby má uživatel možnost připsat příkazy pro tvorbu dalších seznamů – seznam zkratk nebo seznam symbolů.

2.5 Fungování šablony

Zdrojový kód šablony je uložen v souboru `sablona.sty`. Zdrojový kód definuje nové příkazy, nastavuje formátování textu, dokumentu a další. Pro některé definice a úpravy jsou využívány následující externí balíčky, které jsou importovány přímo ve zdrojovém kódu šablony pomocí příkazu `\RequirePackage{}`:

- | | |
|------------------------------|-----------------------------|
| 1. <code>ifthen</code> , | 8. <code>fancyhdr</code> , |
| 2. <code>geometry</code> , | 9. <code>fontspec</code> , |
| 3. <code>lmodern</code> , | 10. <code>hyperref</code> , |
| 4. <code>enumitem</code> , | 11. <code>cite</code> , |
| 5. <code>caption</code> , | 12. <code>etoolbox</code> , |
| 6. <code>subcaption</code> , | 13. <code>babel</code> . |
| 7. <code>chngcntr</code> , | |

Celý soubor je rozdělen do sedmi celků:

1. definice metadat šablony,
2. definice základních příkazů a dat,
3. úvodní textové náležitosti,
4. formátování textu a vzhled stránky,
5. úpravy dalších částí šablony,
6. ostatní pomocné příkazy,
7. multijazykové textové definice.

2.5.1 Definice metadat šablony

Úvodní část zdrojového kódu, která nastavuje číslo verze šablony, datum jejího vydání a název zdrojového souboru (`sablona.sty`). Datum je zadáváno ve formátu `RRRR/MM/DD` (R = rok, M = měsíc, D = den). První zveřejněnou verzí je verze č. 1 s datem vydání 26. 5. 2023.

Příkazy:

- `\def\fileversion{1.0}`,
- `\def\filedate{2023/05/26}`,
- `\def\filename{sablona.sty}`.

Dále je nastaven požadavek na verzi distribuce systému \LaTeX . Nejnovější dostupnou verzí (ve dni vydání šablony) je verze \LaTeX 2 ϵ ze dne 1. 6. 2022. Tuto verzi využívá také šablona pomocí příkazu `\NeedsTeXFormat{LaTeX2e}[2022/06/01]`.

Posledním příkazem úvodní části je příkaz pro představení samotného balíčku. To se vypisuje například do `.log` souboru. Výpis obsahuje název zdrojového souboru, jeho verzi a datum vydání.

Příkaz: `\ProvidesPackage{sablona}[\filedate Šablona v\fileversion]`.

2.5.2 Definice základních příkazů a dat

Tato část zdrojového kódu definuje nové příkazy pro:

1. druh závěrečné práce,
2. název závěrečné práce,
3. odkaz na elektronickou publikaci závěrečné práce,
4. autor závěrečné práce,
5. vedoucí závěrečné práce,
6. ústav,
7. město,
8. rok obhajoby,
9. datum odevzdání,
10. zpracování jazyka.

Využité externí balíčky:

- `ifthen`.

Druh závěrečné práce

Definuje nový příkaz pro zvolení typu závěrečné práce (bakalářská, diplomová nebo dizertační), který uloží do (stejnojmenné) proměnné `prace`. Typ závěrečné práce se zadává zkratkou:

- `bc` = bakalářská práce,
- `ing` = diplomová práce,
- `phd` = dizertační práce.

Kód příkazu:

```
\newcommand\prace[1]{
  \ifthenelse{\equal{\detokenize{#1}}{\detokenize{bc}}}{
    \def\prace{\jazyk@original@prace@bc}{
      \ifthenelse{\equal{\detokenize{#1}}{\detokenize{ing}}}{
        \def\prace{\jazyk@original@prace@ing}{
          \ifthenelse{\equal{\detokenize{#1}}{\detokenize{phd}}}{
            \def\prace{\jazyk@original@prace@phd}{
              \def\prace{\jazyk@original@prace@error}
            }
          }
        }
      }
    }
  }
}
```

Příkaz funguje pomocí balíčku `ifthen`, který poskytuje příkaz `\ifthenelse{}{}{}`. Tento příkaz na základě vyhodnocení podmínky provede dané akce. Příkaz má posloupnost `\ifthenelse{POKUD}{TAK}{JINAK}`.

Kontrola podmínky probíhá porovnáním seznamu znaků pomocí příkazu `\equal{}{}`. Argument příkazu `\prace{}` a zkratky `bc`, `ing` a `phd` jsou převáděny na seznam znaků pomocí příkazu `\detokenize{}`.

Schéma vyhodnocení podmínky příkazu:

Zkontroluj: je zadaná zkratka bc?

Pokud ANO: definuj proměnnou prace a nastav ji na hodnotu
Bakalářská práce

Pokud NE: znovu zkontroluj: je zadaná zkratka ing?

Pokud ANO: definuj proměnnou prace a nastav ji na hodnotu
Diplomová práce

Pokud NE: znovu zkontroluj: je zadaná zkratka phd?

Pokud ANO: definuj proměnnou prace a nastav ji na hodnotu
Dizertační práce

Pokud NE: Vypiš chybovou hlášku

Hodnoty Bakalářská práce, Diplomová práce, Dizertační práce a chybová hláška jsou definovány v proměnných jazyk@original@prace, kterým se věnuje kapitola 2.5.7.

Název závěrečné práce

Definuje příkaz pro zadání názvu závěrečné práce. Název se zadává v plném znění.

Kód příkazu:

```
\newcommand\nazev[1]{
  \def\nazev{#1}
}
```

Příkaz převezme zadaný argument (název závěrečné práce) a uloží jej do (stejnojmenné) proměnné `nazev`.

Odkaz na elektronickou publikaci závěrečné práce

Definuje příkaz pro zadání odkazu na elektronickou publikaci závěrečné práce. Odkaz je vygenerován v systému Studis.

Kód příkazu:

```
\newcommand\odkaz[1]{
  \def\odkaz{#1}
}
```

Příkaz ukládá vložený argument (odkaz na elektronickou publikaci) do proměnné `odkaz`.

Autor závěrečné práce

Definuje příkaz pro zadání jména autora.

Jméno se zadává ve tvaru {tituly před}{jméno}{příjmení}{tituly za}, přičemž jednotlivé tituly (pokud jich je více) se oddělují čárkou a pevnou/nezlomitelnou mezerou (,~). Pokud má autor práce tituly za jménem, příkaz automaticky vloží za příjmení čárku a poté vysází tituly za jménem.

Kód příkazu:

```
\newcommand\autor[4]{
  \ifthenelse{\equal{\detokenize{#4}}{\detokenize{}}}{
```

```

\def\autor@pred{#1}
\def\autor@jmeno{#2}
\def\autor@prijmeni{#3}
\def\autor@za{}}{
  \def\autor@pred{#1}
  \def\autor@jmeno{#2}
  \def\autor@prijmeni{#3}
  \def\autor@za{,~#4.}}
}

```

Příkaz definuje následující čtyři proměnné:

1. `autor@pred` pro tituly před jménem,
2. `autor@jmeno` pro autorovo jméno,
3. `autor@prijmeni` pro autorovo příjmení,
4. `autor@za` pro tituly za jménem.

Pokud autor nemá tituly za jménem, do proměnné `autor@za` je vložen prázdný řetězec, pokud tituly za jménem má, je do této proměnné vložena nejdříve čárka (tituly jsou od příjmení odděleny čárkou), poté nezlomitelná mezera a nakonec tituly.

Kontrola, jestli autor má nebo nemá tituly za jménem, zajišťuje porovnání znakových řetězců. Je-li čtvrtým argumentem příkazu prázdný textový řetězec, je proměnná ukládající tituly za jménem prázdná. Převod argumentu na řetězec znaků poskytuje příkaz `\detokenize{}`.

Schéma kontroly titulů za jménem:

Zkontroluj: je čtvrtý parametr prázdný znakový řetězec?

- Pokud ANO:
1. definuj proměnnou `autor@pred` a ulož do ní tituly před jménem
 2. definuj proměnnou `autor@jmeno` a ulož do ní jméno autora
 3. definuj proměnnou `autor@prijmeni` a ulož do ní autorovo příjmení
 4. definuj proměnnou `autor@za` a ponech ji prázdnou

- Pokud NE:
1. definuj proměnnou `autor@pred` a ulož do ní tituly před jménem
 2. definuj proměnnou `autor@jmeno` a ulož do ní jméno autora
 3. definuj proměnnou `autor@prijmeni` a ulož do ní autorovo příjmení
 4. definuj proměnnou `autor@za` a vlož do ní čárku, nezlomitelnou mezeru a tituly za jménem autora

Vedoucí závěrečné práce

Definuje příkaz pro zadání jména vedoucího závěrečné práce.

Jméno se zadává ve tvaru {tituly před}{jméno}{příjmení}{tituly za}, přičemž jednotlivé tituly (pokud jich je více) se oddělují čárkou a pevnou/nezlomitelnou mezerou.

Kód příkazu:

```
\newcommand\vedouci[4]{
  \ifthenelse{\equal{\detokenize{#4}}{\detokenize{}}}{
    \def\vedouci@pred{#1}
    \def\vedouci@jmeno{#2}
    \def\vedouci@prijmeni{#3}
    \def\vedouci@za{}}{
    \def\vedouci@pred{#1}
    \def\vedouci@jmeno{#2}
    \def\vedouci@prijmeni{#3}
    \def\vedouci@za{,~#4.}}
}
```

Příkaz definuje následující čtyři proměnné:

1. \vedouci@pred pro tituly před jménem,
2. \vedouci@jmeno pro autorovo jméno,
3. \vedouci@prijmeni pro autorovo příjmení,
4. \vedouci@za pro tituly za jménem.

Příkaz funguje analogicky jako příkaz pro zadání jména autora, pouze mění názvy proměnných.

Ústav

Definuje nový příkaz pro volbu ústavu. Ústav se zadává celým názvem.

Kód příkazu:

```
\newcommand\ustav[1]{
  \def\ustav{#1}
}
```

Příkaz vytvoří proměnnou `ustav` a přiřadí jí obsah argumentu z příkazu.

Město

Definuje nový příkaz pro zadání názvu města.

Kód příkazu:

```
\newcommand\mesto[1]{
  \def\mesto{#1}
}
```

Příkaz definuje novou proměnnou `mesto`, které přiřadí hodnotu argumentu příkazu.

Rok obhajoby

Definuje příkaz pro zadání roku obhajoby. Rok se zadává ve formátu RRRR (R = rok).

Kód příkazu:

```
\newcommand\obhajoba[1]{
  \def\obhajoba{#1}
}
```

Příkaz vytváří novou proměnnou `obhajoba`, do které vkládá obsah argumentu příkazu.

Datum odevzdání

Definuje nový příkaz pro zadání data odevzdání závěrečné práce. Datum se zadává ve formátu DD. MM. RRRR (D = den, M = měsíc, R = rok).

Kód příkazu:

```
\newcommand\odevzdani[1]{
  \def\odevzdani{#1}
}
```

Příkaz definuje proměnnou `odevzdani`, které předává svůj argument.

Zpracování jazyka

Definuje nový příkaz pro zadání jazyka. Jazyk zpracování se zadává ve formě zkratky:

- `cz-en` = primární jazyk: čeština, sekundární jazyk: angličtina;
- `sk-en` = primární jazyk: slovenština, sekundární jazyk: angličtina;
- `en-cz` = primární jazyk: angličtina, sekundární jazyk: čeština;
- `en-sk` = primární jazyk: angličtina, sekundární jazyk: slovenština.

V případě, že uživatel zadá špatnou zkratku, případně nezadá žádnou zkratku, bude automaticky nastavena možnost `cz-en`.

Kód příkazu:

```
\newcommand\jazyk[1]{
  \newboolean{logo_EN}
  \ifthenelse{\equal{\detokenize{#1}}{\detokenize{sk-en}}}{
    \jazyk@sk@en}{
    \ifthenelse{\equal{\detokenize{#1}}{\detokenize{en-cz}}}{
      \jazyk@en@cz
      \setboolean{logo_EN}{true}}{
      \ifthenelse{\equal{\detokenize{#1}}{\detokenize{en-sk}}}{
        \jazyk@en@sk
        \setboolean{logo_EN}{true}}{
        \jazyk@cz@en
        }
      }
    }
}
```

Na základě kontroly podmínky volá příkaz jeden z následujících příkazů pro aktivaci multijazykových textových definicí (viz kapitolu 2.5.7):

1. `\jazyk@sk@en`,
2. `\jazyk@en@cz`,
3. `\jazyk@en@sk`,
4. `\jazyk@cz@en`.

Podmínka je kontrolována na shodu znakových řetězců. Argument příkazu a porovnávaná zkratka jsou na znakové řetězce transformovány pomocí příkazu `\detokenize{}`.

Příkaz deklaruje přepínač (boolean) `logo_EN` pro vložení českého či anglického loga fakulty do hlavičky dokumentu. Více v kapitole 2.5.4.

Schéma kontroly zkratky:

Zkontroluj: je zadaná zkratka sk-en?

Pokud ANO: zavolej příkaz `\jazyk@sk@en`

Pokud NE: znovu zkontroluj: je zadaná zkratka en-cz?

Pokud ANO: zavolej příkaz `\jazyk@en@cz`

hodnotu přepínače `logo_EN` přepni na TRUE

Pokud NE: znovu zkontroluj: je zadaná zkratka en-sk?

Pokud ANO: zavolej příkaz `\jazyk@en@sk`

hodnotu přepínače `logo_EN` přepni na TRUE

Pokud NE: zavolej příkaz `\jazyk@cz@en`

2.5.3 Úvodní textové náležitosti

Pro zadání obsahu úvodních textových náležitostí závěrečné práce (abstrakt, klíčová slova, bibliografická citace, čestné prohlášení a poděkování) byly vytvořeny vlastní příkazy. Vytvořené příkazy mají jeden či více argumentů. Popisu těchto argumentů i přednastavení výchozích hodnot se věnuje podkapitola 2.4.2.

Jednotlivé textové náležitosti se sázejí na čistou samostatnou stranu (strana bez záhlaví, zápatí a čísla strany). Každá část začíná vždy na novém listu (tj. na liché straně dokumentu). Příkazy proto během svého běhu volají příkaz `\thispagestyle{empty}` pro zrušení formátování strany (odstranění záhlaví, zápatí a čísla strany) a příkaz pro vytvoření sudé stránky `\cleardoublepage`, který je podrobněji popsán v podkapitole 2.5.5. Výjimku tvoří pouze části abstrakt a klíčová slova, které jsou sázeny společně na jednu stranu.

Nadpisy jednotlivých částí mají stejný formát jako nadpis kapitol. Nejedná se ovšem o kapitoly jako takové, proto je implementován příkaz `\uodninadpis{}`, který formátuje jednotlivé nadpisy těchto částí na formát nadpisu kapitoly a přitom nekládá vstup do obsahu dokumentu. Příkaz je rovněž podrobněji popsán v kapitole 2.5.5.

Abstrakt

Definuje nový příkaz pro zadání abstraktu. Příkaz má dva argumenty pro abstrakt v primárním a sekundárním jazyce.

Kód příkazu:

```
\newcommand\abstrakt[2]{
  \def\abstrakt@original{#1}
  \def\abstrakt@preklad{#2}
  \cleardoublepage
  \thispagestyle{empty}
  \uvodninadpis{\jazyk@original@abstrakt}
  \noindent\abstrakt@original
  \uvodninadpis{\jazyk@preklad@abstrakt}
  \noindent\abstrakt@preklad
}
```

Příkaz deklaruje proměnné `abstrakt@original` a `abstrakt@preklad`, do kterých vkládá text abstraktu v primárním, resp. sekundárním jazyce. Nadpisy v primárním a sekundárním jazyce jsou tvořeny již zmíněným příkazem `\uvodninadpis{}`, kterému jsou předávány proměnné `jazyk@original@abstrakt` a `jazyk@preklad@abstrakt`. Jedná se o proměnné multijazykových textových definicí, kterým se věnuje kapitola 2.5.7.

Příkaz `\noindent` zajišťuje zarovnání textu k levému okraji nehledě na uživatelskou volbu oddělování odstavců. V případě volby oddělení odstavců horizontální zarážkou (viz kapitolu 2.3) tuto zarážku potlačí.

Klíčová slova

Definuje příkaz pro zadání klíčových slov. Příkaz má dva argumenty pro klíčová slova v primárním, resp. sekundárním jazyce.

Kód příkazu:

```
\newcommand\klicovaslova[2]{
  \def\klicovaslova@original{#1}
  \def\klicovaslova@preklad{#2}
  \vfill
  \uvodninadpis{\jazyk@original@klicovaslova}
  \noindent\klicovaslova@original
  \uvodninadpis{\jazyk@preklad@klicovaslova}
  \noindent\klicovaslova@preklad
  \vfill
}
```

Příkaz vytvoří proměnné `klicovaslova@original` a `klicovaslova@preklad`, do kterých předá obsah svých argumentů (klíčová slova v hlavním a vedlejším jazyce). Pomocí multijazykových definicí (viz kapitolu 2.5.7) předá signál příkazu `\uvodninadpis{}` pro tvorbu nadpisů „KLÍČOVÁ SLOVA“ v hlavním jazyce i jazyce vedlejším.

Příkaz `\vfill` zajišťuje rovnoměrné vertikální odsazení od abstraktu a konce stránky.

Bibliografická citace

Definuje nový příkaz pro vysázení bibliografické citace autorovy závěrečné práce. Příkaz

je bez argumentu, využívá již dříve deklarovaných příkazů a jejich dat v následujícím pořadí:

1. jméno a příjmení autora,
2. název závěrečné práce,
3. město,
4. rok obhajoby,
5. odkaz na elektronickou publikaci závěrečné práce,
6. typ závěrečné práce,
7. ústav,
8. jméno a příjmení vedoucího včetně veškerých titulů.

Kód příkazu:

```
\newcommand\citace{
  \cleardoublepage
  \thispagestyle{empty}
  \hbox{}\vfill
  \uvodninadpis{\jazyk@original@citace}
  \noindent
  \MakeUppercase{\autor@prijmeni},~\autor@jmeno. \textit{\nazev}.
  \mesto, \obhajoba. \jazyk@original@citace@odkaz\odkaz. \prace.
  \jazyk@original@univerzita, \jazyk@original@fakulta, \ustav.
  \jazyk@original@citace@vedouci~\vedouci@pred~\vedouci@jmeno~
  \vedouci@prijmeni\vedouci@za
}
```

Příkaz vytvoří nový list pomocí příkazu `\cleardoublepage` a odstraní z něj formátování pomocí příkazu `\thispagestyle{empty}`. Dále vytvoří prázdný box pomocí `\hbox{}`, do kterého pomocí příkazu `\vfill` vloží vertikální mezeru tak, aby bibliografická citace byla umístěna na konci strany.

Samotný nadpis je deklarován v multijazykové proměnné `jazyk@original@citace` v kapitole 2.5.7.

Schéma posloupnosti sázení textové části citace dle normy ČSN ISO 690:

Velkými písmeny vlož příjmení autora, poté jméno.

Kurzívou vlož název práce.

Přidej město a rok obhajoby.

Vlož frázi Dostupné také z: a přidej odkaz.

Vlož typ práce.

Přidej název univerzity, fakulty a ústavu.

Vlož frázi Vedoucí práce: a přidej vedoucího s tituly.

Typografickou sazbu příjmení autora velkými písmeny zajišťuje příkaz `\MakeUppercase{}` a sazbu názvu závěrečné práce poté příkaz `\textit{}`.

Fráze „*Dostupné také z:*“, „*Vedoucí práce:*“ a názvy univerzity a fakulty jsou definovány v kapitole 2.5.7 jako multijazykové proměnné:

- jazyk@original@citace@odkaz,
- jazyk@original@citace@vedouci,
- jazyk@original@univerzita,
- jazyk@original@fakulta.

Čestné prohlášení

Definuje nový příkaz pro zadání čestného prohlášení. Příkaz deklaruje novou proměnnou `prohlaseni`, do které předává svůj argument (textový obsah prohlášení).

Pomocí příkazů `\cleardoublepage` a `\thispagestyle{empty}` je vytvořen nový čistý list. Čestné prohlášení je umístěno na konec strany pomocí příkazů `\hbox{}` a `\vfill`.

Kód příkazu:

```
\newcommand\prohlaseni[1]{
  \def\prohlaseni{#1}
  \cleardoublepage
  \thispagestyle{empty}
  \hbox{}\vfill
  \uvodninadpis{\jazyk@original@prohlaseni}
  \noindent\prohlaseni
  \par\vspace{2cm}
  \noindent\jazyk@original@podpis~\odevzdani\hfill
  \makebox[6cm]{\dotfill}
  \makebox[\textwidth]{\hfill\makebox[6cm][c]{\autor@pred~\autor@jmeno~
  \autor@prijmeni\autor@za}}
}
```

Za textovým obsahem prohlášení následuje datum odevzdání a prostor pro autorův podpis. To je od textu prohlášení odděleno vertikální mezerou o velikosti 2 cm pomocí příkazu `\vspace{2cm}`.

Příkaz využije multijazykové proměnné `jazyk@original@podpis` pro vložení textové fráze „V Brně dne“, za kterou následuje datum odevzdání. Následně příkaz na stejném řádku vytvoří horizontální mezeru `\hfill` a vytvoří box dlouhý 6 cm `\makebox[6cm]{}`, který vyplní tečkami `\dotfill`.

Na novém řádku vytvoří nový box o délce jednoho řádku `\makebox[\textwidth]{}`. Do tohoto boxu vloží horizontální odsazení `\hfill` a nový box dlouhý 6 cm se zarovnáním na střed `\makebox[6cm][c]{}`. Do tohoto boxu následně vloží jméno autora závěrečné práce včetně jeho titulů.

Poděkování

Vytvoří nový příkaz pro zadání poděkování. Příkaz definuje proměnnou `podekovani`, které předává obsah svého argumentu.

Kód příkazu:

```
\newcommand\podekovani[1]{
  \def\podekovani{#1}
  \cleardoublepage
  \thispagestyle{empty}
  \hbox{}\vfill
  \uvodninadpis{\jazyk@original@podekovani}
  \noindent\podekovani
}
```

Pomocí příkazů `\cleardoublepage`, `\thispagestyle{empty}`, `\hbox{}` a `\vfill` je vytvořen nový list a textový obsah poděkování je umístěn na konec liché stránky.

2.5.4 Formátování textu a vzhled stránky

Tato část nastavuje vzhled dokumentu prostřednictvím následujících položek:

1. formátování rozměrů stránky,
2. formátování písma a odstavců,
3. formátování nadpisů kapitol,
4. formátování popisků obrázků a tabulek,
5. formát číslování obrázků, tabulek a rovnic,
6. hlavička a patička dokumentu.

Využité externí balíčky:

- | | |
|-------------|---------------|
| - geometry, | - subcaption, |
| - lmodern, | - chngcntr, |
| - enumitem, | - fancyhdr, |
| - caption, | - fontspec. |

Formátování rozměrů stránky

Směrnice děkana FSI č. 3/2022 stanovuje rozměry textové oblasti na 16 cm × 25 cm. Tomu odpovídá velikost okrajů:

- vlevo/u hřbetu 3 cm (kvůli vazbě),
- vpravo 2 cm,
- nahoře 2,2 cm,
- dole 2,5 cm.

Pro nastavení okrajů stránky je využíván balíček `geometry`.

Kód příkazu:

```
\geometry{
  left=3cm,
  top=2.2cm,
  text={16cm,25cm},
  footskip=1cm
}
```

Příkaz má následující parametry:

- `left` pro nastavení velikosti levého okraje,
- `top` pro nastavení velikosti horního okraje,
- `text` pro nastavení velikosti textové oblasti,
- `footskip` pro nastavení velikosti mezery mezi koncem textové oblasti a číslem strany.

Formátování písma a odstavců

Nastavuje výchozí písmo pro celý dokument, jeho velikost a řádkování. Zároveň nastavuje mezery mezi odstavci a definuje novou uživatelskou volbu pro oddělování odstavců (viz kapitolu 2.3).

Pomocí příkazu `\usepackage{lmodern}` je typ písma změněn na `Latin Modern Roman`.

Směrnice děkana č. 3/2022 dále stanovuje velikost řádkování na 1 až 1,1. Šablona pomocí příkazu `\renewcommand{\baselinestretch}{1.1}` využívá řádkování o velikosti 1,1.

Pro formátování odstavců je vytvořena uživatelská volba popsaná v kapitole 2.3. Uživateli dává na výběr ze dvou možností:

1. Nový odstavec bude začínat odstavcovou zarážkou, tj. první slovo nového odstavce bude zaraženo o 1 cm doprava.
2. Nový odstavec bude bez odstavcové zarážky a od předchozího odstavce bude odsazen o 6 pt ve vertikálním směru.

Aby bylo zachováno stejné odsazení i v seznamech (s odrážkami nebo čísly), jsou pomocí balíčku `enumitem` nastaveny mezery před a za seznam na 0 pt, resp. 6 pt v závislosti na volbě dělení odstavců.

Kód příkazu:

```
\RequirePackage{enumitem}
\newcommand\odstavec[1]{
  \ifthenelse{#1 = 1}{
    \setlength{\parindent}{1cm}
    \kapitoly@volba@prvni
    \setlist{
      topsep = 6pt,
      partopsep = 0 pt,
      noitemsep
    }
  }{
    \setlength{\parindent}{0cm}
    \setlength{\parskip}{6pt}
    \kapitoly@volba@druha
    \setlist{nosep}
  }
}
```

Schéma kontroly uživatelské volby:

Zkontroluj: zadal uživatel volbu č. 1?

Pokud ANO: 1. nastav odstavcovou zarážku `\parindent` na 1 cm
 2. aktivuj formátování nadpisů kapitol
 3. nastav mezery před a za seznamy na 6 pt a
 smaž vertikální mezery uvnitř seznamů

Pokud NE: 1. nastav odstavcovou zarážku `\parindent` na 0 cm
 2. nastav mezeru mezi odstavci `\parskip` na 6 pt
 3. aktivuj formátování nadpisů kapitol
 4. potlač vertikální mezery uvnitř seznamů

Parametr `\parskip` nastavuje vertikální mezery mezi odstavci v celém dokumentu. K této vertikální mezeře se poté přičítají jednotlivá odsazení nadpisů kapitol a podobně.

Z tohoto důvodu je nutné provést dvojí nastavení formátu nadpisů kapitol pomocí příkazů `\kapitoly@volba@prvni` a `\kapitoly@volba@druha`.

Formátování nadpisů kapitol

Definuje nové příkazy `\kapitoly@volba@prvni` a `\kapitoly@volba@druha`. Oba příkazy mají stejnou definici – nastavují formátování nadpisů kapitol. Jak je již zmíněno v předchozích odstavcích, je nutné respektovat parametr `\parskip`. Na následujících řádcích jsou ukázány části kódu pro jednotlivé typy nadpisů (na základě volby dělení odstavců č. 1, viz kapitolu 2.3).

Kapitola úrovně 1 (X):

Příkaz: `\chapter{}`.

- začíná vždy na nové straně,
- kapitálky,
- tučný text,
- velikost písma 16 pt,
- odsazení za nadpisem 24 pt.

Kód nastavení:

```
\renewcommand\chapter{
  \cleardoublepage
  \@startsection{chapter}{0}{0pt}
    {-1ex}{24pt}
    {\fontsize{16}{19.2}\bfseries\MakeUppercase}
}
```

Příkaz `\@startsection` je vnitřní příkaz systému \LaTeX , který šablona pouze modifikuje. Jeho syntaxe je následující:

```
\@startsection{název sekce}{úroveň}{odsazení od levého okraje}
  {odsazení před nadpisem}{odsazení po nadpisu}
  {styl}
```

Systém \LaTeX čísluje úrovně kapitol od 0, proto je na pozici `{úroveň}` uvedeno číslo 0.

Příkazy `\fontsize{16}{19.2}\bfseries\MakeUppercase` nastavují velikost písma na 16 pt, tučný text a kapitálky.

Kapitola úrovně 2 (X.X):

Příkaz: `\section{}`.

- tučný text,
- velikost písma 14 pt,
- odsazení před nadpisem 24 pt,
- odsazení za nadpisem 12 pt.

Kód nastavení:

```
\renewcommand\section{
  \@startsection{section}{1}{0pt}
    {-24pt}{12pt}
    {\fontsize{14}{16.8}\bfseries}
}
```

Kapitola úrovně 3 (X.X.X):

Příkaz: `\subsection{}`.

- tučný text,
- velikost písma 13 pt,
- odsazení před nadpisem 12 pt,
- odsazení za nadpisem 6 pt.

Kód nastavení:

```
\renewcommand\subsection{
  \@startsection{subsection}{2}{0pt}
    {-12pt}{6pt}
    {\fontsize{13}{15.6}\bfseries}
}
```

Kapitoly úrovně 4 (X.X.X.X) a další se běžně nepoužívají, nejsou číslovány a nejsou zobrazovány v obsahu. Z tohoto důvodu nejsou zahrnuty ani v této šabloně.

Pro volbu dělní odstavců č. 2 je nastavení formátu nadpisů analogické, pouze zohledňuje mezeru 6 pt danou parametrem `parskip`.

Formátování popisků obrázků a tabulek

Pomocí balíčků `caption` a `subcaption` je možné nastavit odsazení kolem obrázků a také kolem popisků obrázků. Mezery před a za obrázkem jsou nastaveny pomocí příkazu `\setlength\intextsep{18pt}` na velikost 18 pt.

Popisek bývá umístěn pod obrázkem, je oddělen mezerou o velikosti 6 pt a od následujícího odstavce mezerou o velikosti 12 pt příkazy:

- `\setlength\abovecaptionskip{6pt}`,
- `\setlength\belowcaptionskip{12pt}`.

Popisek tabulky se zpravidla umísťuje nad tabulku a je zarovnan doleva. Zarovnání doleva poskytuje příkaz `\captionsetup[table]{singlelinecheck=false}`.

Formát číslování obrázků, tabulek a rovnic

Definuje nový příkaz pro volbu číslování. Uživatel má na výběr dvě možnosti:

1. Číslo obsahuje i číslo hlavní kapitoly, tzn. číslo bude ve formátu X.X: Popisek.
2. Číslo obsahuje pouze pořadové číslo, tzn. bude ve formátu X: Popisek.

Kód příkazu:

```
\newcommand\cislovani[1]{
  \ifthenelse{#1 = 1}{
    \counterwithin{equation}{chapter}
    \counterwithin{figure}{chapter}
    \counterwithin{table}{chapter}
  }{
    \counterwithout{equation}{chapter}
    \counterwithout{figure}{chapter}
    \counterwithout{table}{chapter}
  }
}
```

Pro nastavení číslování je využíván balíček `chngcntr`, který má implementovanu možnost pro nastavení číslování obrázků, tabulek a rovnic pomocí příkazu `\counterwithin{}{}` a `\counterwithout{}{}`. První argument obou příkazů je typ počítadla (obrázek, tabulka, rovnice), druhý argument je poté sekce, kterou má či nemá číslo obsahovat.

Hlavička a patička dokumentu

Tato část nastavuje formát hlavičky a patičky dokumentu. Hlavička je rozdílná na sudé a liché straně. Lichá strana obsahuje logo fakulty (v české nebo anglické verzi), název ústavu, město a rok obhajoby. Sudá strana poté logo VUT, jméno autora (včetně titulů) a typ závěrečné práce.

Aby bylo možné upravovat záhlaví a zápatí, je potřeba nastavit styl stránky pomocí `\pagestyle{fancy}`.

Hlavička využívá možnost balíčku `fontspec` pro změnu fontu písma, který není definován v systému \LaTeX . Použitý font je převzatý z titulní strany a jedná se o VUT font **Vafle** v provedení **Light**. Zdrojový soubor tohoto fontu (dostupný po přihlášení na vut.cz/jvs) je uložen v hlavním adresáři šablony ve složce **Fonty**.

Pro aktivaci tohoto fontu je deklarován nový příkaz `\fontVUT`:

```
\newcommand\fontVUT{
  \fontspec[
    Path = fonty/,
    Extension = .otf
  ]{Font_VUT_Vafle_Light}
  \fontsize{10pt}{11pt}
```



```
\selectfont
}
```

Pomocí příkazu `\fontspec[]{}{}` jsou nejdříve načtena specifika vybraného fontu. Cesta ke zdrojovému souboru fontu je systému \LaTeX ukázána prostřednictvím parametru `Path`, přípona souboru pomocí parametru `Extension` a následně je ve složených závorkách uveden název zdrojového souboru fontu, pod kterým je v daném adresáři uložen. Příkaz `\fontsize{}{}{}` následně nastaví velikost písma na 10 pt a aktivace fontu probíhá příkazem `\selectfont`.

Pro vložení loga fakulty a univerzity do hlavičky liché, resp. sudé stránky byly implementovány dva pomocné příkazy:

1. `\logoFSI` pro logo fakulty,
2. `\logoVUT` pro logo VUT.

Příkaz `\logoFSI` využívá již dříve vytvořeného přepínače (boolean) `logo_EN` (viz kapitolu 2.5.2).

Kód příkazu:

```
\newcommand\logoFSI{
  \ifthenelse{\boolean{logo_EN}}{
    \raisebox{-1.5pt}{\includegraphics[height=16pt]
      {loga/Logo_FSI_EN.png}}}{
    \raisebox{-1.5pt}{\includegraphics[height=16pt]
      {loga/Logo_FSI_CZ.png}}
  }
}
```

Příkaz `\boolean{logo_EN}` otestuje, zdali je přepínač `logo_EN` nastaven na hodnotu `TRUE`. Pokud ano, do příkazu `\logoFSI` importuje anglickou verzi loga fakulty. Pokud přepínač nebo na hodnotě `TRUE`, bude importována česká verze loga. V obou případech je výška loga nastavena na velikost 16 pt a umístění loga posunuto o 1,5 pt směrem vzhůru pomocí příkazu `\raisebox{-1.5pt}`.

Logo VUT je stejné v anglickém i českém jazyce, proto není potřeba kontrolovat primární jazyk závěrečné práce.

Příkaz `\logoVUT`:

```
\newcommand\logoVUT{
  \raisebox{-1.5pt}{\includegraphics[height=16pt]{loga/Logo_VUT.png}}
}
```

Stejně jako v případě loga fakulty je velikost VUT loga nastavena na 16 pt a posunutí nastaveno na 1,5 pt směrem vzhůru.

Pole hlavičky má výšku 20 pt, kterou nastavuje příkaz `\setlength{\headheight}{20pt}`. Hlavička je od textové oblasti oddělena horizontální čarou o tloušťce 0,4 pt. Toto zajišťuje příkaz `\renewcommand{\headrulewidth}{0.4pt}`. V případě patičky je oddělovací čára skryta pomocí příkazu `\renewcommand{\footrulewidth}{0pt}`.

Typografická sazba hlavičky je prováděna pomocí sekvence příkazů:

```
\fancyhead{}
\fancyhead[L0]{\logoVUT\hspace{0.5em}\fontVUT\autor@pred~\autor@jmeno~
               \autor@prijmeni\autor@za}
\fancyhead[R0]{\fontVUT\prace}
\fancyhead[LE]{\logoFSI\hspace{0.5em}\fontVUT\ustav}
\fancyhead[RE]{\fontVUT\mesto,~\obhajoba}
```

Příkaz `\fancyhead{}` nejdříve vyčistí pole hlavičky a poté do něj začne vkládat jednotlivé položky. V hranatých závorkách se střídá několik parametrů. Jedná se o anglické zkratky polohy:

- L = left umístění vlevo,
- R = right umístění vpravo.

A anglické zkratky sudé a liché strany:

- O = odd page lichá strana,
- E = even page sudá strana.

Umístění položek v hlavičce:

- [L0] logo VUT, horizontální mezera 0,5 em, VUT fontem jméno a příjmení autora včetně titulů;
- [R0] VUT fontem typ práce;
- [LE] logo FSI, horizontální mezera 0,5 em, VUT fontem název ústavu;
- [RE] VUT fontem město a rok obhajoby.

V patičce je na pozicích [LE,R0] pouze zobrazeno číslo strany pomocí `\thepage`.

Kód tvorby patičky:

```
\fancyfoot{}
\fancyfoot[LE,R0]{\thepage}
```

2.5.5 Úpravy dalších částí šablony

V tomto celku probíhá úprava zobrazení obsahu, nastavení křížových odkazů uvnitř dokumentu, předání dat PDF prohlížeči a aktivace citačního balíčku.

Využité externí balíčky:

- hyperref,
- cite,
- etoolbox.

Úprava zobrazení obsahu

V případě volby dělení odstavců možností č. 2 (viz kapitolu 2.3) docházelo k vkládání odstavcové mezery mezi jednotlivé vstupy kapitol v obsahu. Z toho důvodu byl vytvořen příkaz `\obsah`, který využívá standardního L^AT_EX příkazu `\tableofcontents` pro tvorbu obsahu, ale ruší u něj vkládání odstavcové mezery.

Kód příkazu:

```
\newcommand\obsah{{  
  \setlength{\parskip}{0pt}  
  \tableofcontents  
}}
```

Příkaz využívá dvojité složené závorky `{{...}}`, aby vyrušení odstavcové mezery proběhlo pouze v rámci prostředí obsahu.

Křížové odkazy a PDF metadata

Dochází k nastavení křížových odkazů v rámci dokumentu. Při prohlížení PDF dokumentu je možné v obsahu kliknout na kapitolu a prohlížeč uživatele automaticky přesune na vybranou kapitolu. Totéž funguje i při kliknutí na citaci, číslo obrázku nebo kapitolu, na kterou uživatel během textu odkazuje. Vytvoření křížového odkazu je možné pomocí dvojice příkazů `\label{NAZEV_ODKAZU}` a `\ref{NAZEV_ODKAZU}`. Příkaz `\label{}` vytvoří místo, na které bude následně příkazem `\ref{}` odkazováno. Název odkazu je volitelný.

Předání dat PDF prohlížeči:

```
\AtBeginDocument{  
  \hypersetup{  
    pdftitle={\nazev},  
    pdfauthor={\autor@pred~\autor@jmeno~\autor@prijmeni\autor@za},  
    pdfsubject={\prace},  
  }  
}
```

Na začátku tvorby dokumentu (`\AtBeginDocument{}`) se pomocí příkazu `\hypersetup{}` předají PDF prohlížeči následující údaje, které prohlížeč zobrazí ve vlastnostech dokumentu:

1. `pdftitle` název závěrečné práce,
2. `pdfauthor` jméno autora včetně jeho titulů,
3. `pdfsubject` předmět práce (typ závěrečné práce).

Další nastavení je implementováno kódem:

```
\hypersetup{  
  pdfstartview={},  
  pdfdisplaydoctitle=true,  
  baseurl=\odkaz,  
  colorlinks=true,  
  allcolors=black  
}
```

Význam jednotlivých parametrů:

- `pdfstartview={}` respektuje nastavení zobrazení PDF prohlížeče, tj. nepřepisuje jej,

- `pdfdisplaydoctitle=true` zajišťuje zobrazení názvu závěrečné práce ve stavové liště prohlížeče místo názvu souboru, pod kterým je dokument uložen,
- `baseurl=\odkaz` předává prohlížeči odkaz na elektronickou publikaci závěrečné práce,
- `colorlinks=true` umožňuje nastavit barvu křížových i webových odkazů,
- `allcolors=black` nastavuje barvu všech odkazů (křížových i webových) na černou.

Citační balíček

Sekce pomocí příkazu `\RequirePackage{cite}` aktivuje citační balíček `cite` a následně mu pomocí příkazu `\bibliographystyle{czplain}` přiřadí styl citování `czplain`.

Stylový balíček `czplain` pochází ze šablony závěrečných prací na Fakultě informačních technologií VUT a autorem této šablony byl částečně modifikován z důvodu změny citační normy ČSN EN ISO 690 na konci roku 2022. Zdrojový soubor tohoto stylového balíčku je možné stáhnout na webových stránkách FIT: fit.vut.cz/study/theses/master-theses/.cs.

Prostředí verbatim

Prostředí `verbatim` je využíváno pro zobrazení kódu v textu. Text napsaný uvnitř prostředí `verbatim` je zobrazován tak, jak je napsán, tj. žádné příkazy uvnitř tohoto prostředí nejsou překládány (nevykonávají svou funkci). Všechny ukázky kódů příkazů v tomto dokumentu jsou psány právě v prostředí `verbatim`.

Balíček `etoolbox` umožňuje upravit odsazení před a za prostředím `verbatim`. Toho je docíleno pomocí příkazu `\preto{\@verbatim}{\topsep=0pt \partopsep=0pt}`, který mezery před a za tímto prostředím nastavuje na velikost 0 pt.

2.5.6 Ostatní pomocné příkazy

Krátký celek obsahující pomocné příkazy pro tvorbu prázdné dvojstránky, resp. prázdné sudé stránky a pro tvorbu nadpisů úvodních textových náležitostí (viz podkapitolu 2.5.3).

Prázdná sudá stránka

Upravuje vnitřní příkaz systému \LaTeX `\cleardoublepage`. Při použití příkazu je kontrolováno, zdali je dokument tisknut oboustranně a zdali se dokument nachází na liché straně. Pokud ano, zapamatuje si její číslo. Pokud ne, je vytvořena prázdná sudá strana (tj. strana bez záhlaví a zápatí).

Kód příkazu:

```
\renewcommand\cleardoublepage{
  \clearpage
  \if@twoside
    \ifodd
      \c@page
    \else
      \hbox{}
      \thispagestyle{empty}
    \newpage
```

```

\fi
\fi
}

```

Pro kontrolu oboustranného tisku je využívána kontrola systému L^AT_EX `\if@twoside` a pro kontrolu liché strany příkaz `\ifodd`.

Úvodní nadpis

Jak již bylo zmíněno v podkapitole 2.5.3, pro úvodní textové náležitosti (abstrakt, klíčová slova, bibliografická citace, prohlášení autora a poděkování) byl implementován vlastní příkaz pro formátování nadpisů těchto částí.

Formátování nadpisu:

- kapitálky,
- tučný text,
- velikost písma 16 pt,
- odsazení před nadpisem 24 pt,
- odsazení za nadpisem 6 pt.

Kód příkazu:

```

\newcommand\uvodninadpis[1]{
  \par\vspace{24pt}\noindent
  {\fontsize{16}{19.2}\bfseries\MakeUppercase{#1}}
  \par\vspace{6pt}
}

```

2.5.7 Multijazykové textové definice

Na základě volby jazyka zpracování definuje jednotlivé textové řetězce používané v rámci dokumentu.

Využité externí balíčky:

- `babel`.

V této části jsou definovány jednotlivé příkazy volané kontrolou volby jazyka zpracování z kapitoly 2.3:

- `\newcommand\jazyk@cz@en{}` pro česko-anglický dokument,
- `\newcommand\jazyk@sk@en{}` pro slovensko-anglický dokument,
- `\newcommand\jazyk@en@cz{}` pro anglicko-český dokument,
- `\newcommand\jazyk@en@sk{}` pro anglicko-slovenský dokument.

Seznam jednotlivých textových řetězců:

1. Název univerzity a fakulty:
 - `\def\jazyk@original@univerzita{Vysoké učení technické v Brně},`
 - `\def\jazyk@original@fakulta{Fakulta strojního inženýrství}.`
2. Typ práce:
 - `\def\jazyk@original@prace@bc{Bakalářská práce},`

- \def\jazyk@original@prace@ing{Diplomová práce},
- \def\jazyk@original@prace@phd{Dizertační práce},
- \def\jazyk@original@prace@error{Nezadán/neznámý typ práce}.

3. Úvodní textové náležitosti a jejich překlad:

- \def\jazyk@original@abstrakt{Abstrakt},
 - \def\jazyk@preklad@abstrakt{Abstract},
- \def\jazyk@original@klicovaslova{Klíčová slova},
 - \def\jazyk@preklad@klicovaslova{Keywords},
- \def\jazyk@original@citace{Bibliografická citace},
 - \def\jazyk@original@citace@odkaz{Dostupné také z: },
 - \def\jazyk@original@citace@vedouci{Vedoucí práce:},
- \def\jazyk@original@prohlaseni{Čestné prohlášení},
- \def\jazyk@original@podpis{V Brně dne},
- \def\jazyk@original@podekovani{Poděkování}.

Textové řetězce ostatních jazykových mutací jsou totožné, pouze jsou přeloženy do příslušného jazyka.

3 ZHODNOCENÍ A DISKUZE

Při tvorbě šablony byl kladen důraz na jednoduchost a použitelnost. Cílem autora bylo vytvořit šablonu a popsat její jednotlivé části tak, aby bylo možné pochopit její fungování i bez větších či hlubších znalostí systému \LaTeX . Veškeré vytvořené příkazy jsou proto doplněny o popisky jak s nimi pracovat, co je možné zadávat do jejich případných argumentů a jaké jsou jejich přednastavené hodnoty.

Šablona je adekvátním nástrojem pro psaní závěrečných vysokoškolských kvalifikačních prací na Fakultě strojního inženýrství Vysokého učení technického v Brně. Vzhledem k podrobným komentářům a jednotné struktuře zdrojového kódu je šablona přehlednější než některé z využívaných šablon na VUT. Šablona je tak více otevřenější pro další rozvoj a rozšíření na další fakulty.

Známé problémy

Při vytváření nečíslovaných kapitol (úvod, závěr, literatura a seznamy) nedochází k automatickému vytváření kotvy pro křížové odkazy. To je vyřešeno pomocí vložení imaginární sekce `\phantomsection`, která tuto kotvu vytvoří. Křížový odkaz ovšem chybně odkazuje na konec dané kapitoly, nikoliv na její začátek.

Kvůli tvorbě nečíslovaných kapitol a jejich ručnímu vkládání do obsahu byla zjištěna nekompatibilita s externím balíčkem `amsmath`, který rozšiřuje možnosti práce s matematickými rovnicemi. Při pokusu o překlad s tímto aktivovaným balíčkem vyskočí chybová hláška „*LaTeX Error: Something's wrong—perhaps a missing \item*“. Jedná se o obecnou chybovou hlášku systému \LaTeX , která přímo neukazuje na objevený problém. Při prohledávání `.log` souboru se tato chybová hláška objevuje v místě vkládání nečíslovaných kapitol do obsahu. Za balíčkem stojí sdružení AMS – American Mathematical Society, kterému byl tento problém nahlášen, a lze proto předpokládat, že tato chyba bude opravena v některé z budoucích aktualizací balíčku.

Možnosti dalšího rozvoje

Dalším rozvojem (například v rámci diplomové práce) je možné šablonu rozšířit o využití i pro semestrální práce. Při zadání typu práce jako „semestrální“ by šablona automaticky vyřadila zpracování úvodních textových náležitostí (abstrakt, klíčová slova, bibliografická citace, čestné prohlášení a poděkování) a případně také seznamy použité literatury, obrázků, tabulek a dalších.

Již komplexnějším rozvojem se jeví možnost tvorby nového stylového balíčku pro tvorbu citací tak, aby splňoval veškeré náležitosti citační normy po její aktualizaci na konci roku 2022 a aby tento stylový balíček byl přímou součástí zdrojového kódu šablony.

ZÁVĚR

Oblíbenost systému \LaTeX společně s jeho nesporně vysokou kvalitou typografické sazby dokumentů dala vzniknout šabloně závěrečných prací pro Fakultu strojního inženýrství na Vysokém učení technickém v Brně. Šablona je navržena s ohledem na kvalitativní požadavky vysokoškolských kvalifikačních prací a zároveň respektuje pokyny stanovené Směrnicí děkana č. 3/2022, která je doplňkem Směrnice rektora č. 72/2017 „Úprava, odevzdávání a zveřejňování závěrečných prací“.

Šablona byla vytvořena s ohledem na jednoduchost a obsahuje podrobné popisy používaných příkazů včetně příkladů použití v textu práce i přímo ve zdrojovém kódu.

Jádrem celé šablony jsou tři pracovní soubory – nastavení, hlavní pracovní soubor a stylový balíček.

Soubor nastavení poskytuje prostor s vlastními příkazy pro vyplnění základních údajů o závěrečné práci, jako jsou název práce, její typ, jméno autora a vedoucího, jazyk zpracování a další. Zároveň nabízí možnost volby druhu oddělování odstavců a typu číslování obrázků, tabulek a rovnic.

Hlavní pracovní soubor je soubor, který řídí tvorbu celého dokumentu. Uživatel v něm aktivuje samotnou šablonu, vepisuje úvodní textové náležitosti (abstrakt, klíčová slova, bibliografickou citaci a další), importuje titulní stranu, zadání a jednotlivé kapitoly.

Stylový balíček je zdrojový soubor šablony, který obsahuje definici všech vlastních příkazů vytvořených pro potřeby šablony. Příkazy upravují formální i funkční stránku šablony. K tomu jsou využívány i externí balíčky, které rozšiřují základní funkcionality systému \LaTeX .

Pro přehlednost jsou v adresáři se šablonou vytvořeny podadresáře pro třídění jednotlivých souborů (kapitoly, přílohy, obrázky, písma a další), které zpřehledňují práci se šablonou.

Tato závěrečná práce obsahuje také úvodní rešeršní kapitolu s krátkým přehledem nejpožívanějších \LaTeX editorů s jejich popisem a shrnutím výhod a nevýhod včetně doporučení vhodných editorů pro široké využití.

V části vlastního řešení se následně jednotlivé kapitoly věnují popisu všech tří pracovních souborů, kde jsou ukázány všechny vytvořené příkazy, jak vypadají, jak fungují, jaké mají přednastavené hodnoty a jak se příkazy zachovají, pokud budou uživatelem nesprávně použity.

Šablona poskytuje jednoduchý nástroj pro tvorbu závěrečných prací s potenciálem stát se nejenom jednotnou fakultní šablonou, ale také jednotnou šablonou pro celé Vysoké učení technické v Brně.

LITERATURA

- [1] KOPKA, H. a DALY, P. W. *LATEX: podrobný průvodce*. 1. vyd. Brno: Computer Press, 2004. ISBN 80-7226-973-9.
- [2] RYBIČKA, J. *Latex pro začátečníky*. 3. vyd. Brno: Konvoj, 2003. ISBN 80-7302-049-1.
- [3] SNIGDHA. 13 Best LaTeX Editors You Must Try in 2023. *TechPout*. Online. Jaipur: TechPout Products and Services, c2023. Dostupné z: <https://www.techpout.com/best-latex-editors/>. [Citováno 2023-03-07].
- [4] MARTIN, M. 14 BEST LaTeX Editor for Mac & Windows in 2023. *Guru99*. Online. Wilmington: Guru99, c2023. Dostupné z: <https://www.guru99.com/best-latex-editors-window-mac.html>. [Citováno 2023-03-07].
- [5] A.N., S. 12 Best LaTeX Editors You Should Use. *Beebom*. Online. Nové Dillí: Beebom Media Private Limited. Dostupné z: <https://beebom.com/best-latex-editors/>. [Citováno 2023-03-07].
- [6] BRACHET, P. *TEXMAKER: Free cross-platform LaTeX editor since 2003*. Online. Agen: Pascal Brachet, c2003-2023. Dostupné z: <https://www.xmlmath.net/texmaker/index.html>. [Citováno 2023-02-24].
- [7] Welcome to T_EXstudio. *T_EXstudio - A L^AT_EX editor*. Online. Lübeck: Benito van der Zander. Dostupné z: <https://www.texstudio.org>. [Citováno 2023-03-02].
- [8] Features. *T_EXstudio - A L^AT_EX editor*. Online. Lübeck: Benito van der Zander. Dostupné z: <https://www.texstudio.org>. [Citováno 2023-03-02].
- [9] What is LyX? *LyX – The Document Processor*. Online. The LyX Team. Dostupné z: <https://www.lyx.org/WhatIsLyX>. [Citováno 2023-03-06].
- [10] Ukázka LyX editoru. *LyX – The Document Processor*. Online. The LyX Team. Dostupné z: <https://www.lyx.org/Screenshots>. [Citováno 2023-03-06].
- [11] Features. *LyX – The Document Processor*. Online. The LyX Team. Dostupné z: <https://www.lyx.org/Features>. [Citováno 2023-03-06].
- [12] About us. *Overleaf*. Online. Londýn: Overleaf c/o Digital Science, c2023. Dostupné z: <https://www.overleaf.com/about>. [Citováno 2023-03-02].
- [13] Overleaf premium features. *Overleaf*. Online. Londýn: Overleaf c/o Digital Science, c2023. Dostupné z: https://www.overleaf.com/learn/how-to/Overleaf_premium_features. [Citováno 2023-03-02].
- [14] Choose Your Plan. *Overleaf*. Online. Londýn: Overleaf c/o Digital Science, c2023. Dostupné z: <https://www.overleaf.com/user/subscription/plans>. [Citováno 2023-03-02].

- [15] What is the maximum compilation time, file number and project size allowed on free vs paid plans? *Overleaf*. Online. Londýn: Overleaf c/o Digital Science, c2023. Dostupné z: https://www.overleaf.com/learn/how-to/What_is_the_maximum_compilation_time%2C_file_number_and_project_size_allowed_on_free_vs_paid_plans%3F. [Citováno 2023-04-03].
- [16] *Papeeria*. Praha: BarD Software, 2018. Dostupné z: <https://www.papeeria.com/landing>. [Citováno 2023-03-10].
- [17] Pricing Plans. *Papeeria*. Online. Praha: BarD Software, 2020. Dostupné z: <https://www.papeeria.com/about/pricing>. [Citováno 2023-03-10].
- [18] Ukázka prostředí Papeeria editoru. *Papeeria*. Online. Praha: BarD Software, 2018. Dostupné z: <https://www.papeeria.com/landing>. [Citováno 2023-03-10].

SEZNAM OBRÁZKŮ

1	Logo systému \LaTeX	17
2	Ukázka prostředí \TeX maker editoru.	18
3	Ukázka prostředí \TeX studio editoru.	19
4	Ukázka prostředí LyX editoru.	20
5	Ukázka prostředí Overleaf editoru.	21
6	Ukázka prostředí Papeeria editoru.	22

A PŘÍLOHA Č. 1 – SOUBOR ČTI MĚ

Pro překlad použít Lua \LaTeX a BIB \TeX .

Šablona závěrečné práce je tvořena následujícími soubory:

1. `nastaveni.tex` – soubor pro zadání základních údajů o závěrečné práci (druh závěrečné práce, název, odkaz na elektronickou publikaci, autor, vedoucí, ústav, město, rok obhajoby, datum odevzdání a jazyk zpracování). Dále je nastavováno formátování odstavců a číslování obrázků, tabulek a rovnic.
2. `zaverecna_prace.tex` – hlavní pracovní soubor, který se překládá překladačem Lua \LaTeX a BIB \TeX . V souboru se aktivuje samotná šablona, aktivují se volitelné externí balíčky, vkládá se titulní strana a zadání. Do tohoto souboru uživatel také přímo vepisuje úvodní textové náležitosti (abstrakt, klíčová slova, prohlášení a poděkování), vkládá jednotlivé kapitoly a probíhá generování obsahů.
3. `sablona.sty` – balíček, zdrojový kód šablony.
4. `citace.bib` – soubor pro vkládání zdrojového textu použité literatury pro program BIB \TeX .
5. `czplain.bst` – stylový balíček pro program BIB \TeX , který upravuje styl citací dle ČSN ISO 690.
6. `zaverecna_prace.pdf` – výsledný PDF dokument celé závěrečné práce.

Pomocné adresáře:

1. `fonty` – složka se zdrojovými soubory VUT fontů použitých v šabloně.
2. `kapitoly` – složka pro ukládání souborů ve formátu `.tex` s textovým obsahem jednotlivých kapitol. Kapitoly ÚVOD a ZÁVĚR pojmenovat jako `00_uvod.tex` a `00_zaver.tex`.
3. `loga` – složka s logy fakulty a univerzity, která jsou použita v šabloně.
4. `obrazky` – složka pro ukládání obrázků použitých v textu práce.
5. `soubory` – složka pro titulní stranu a zadání ve formátu `.pdf`. Soubory pojmenovat jako `titulni_strana.pdf` a `zadani.pdf`.

Soubory s příponami `.aux`, `.bbl`, `.blg`, `.lof`, `.log`, `.lot`, `.out` a `.toc` jsou soubory vytvářené systémem \LaTeX během překladu zdrojového textu. Do těchto souborů si systém ukládá potřebná data používaná pro generování výsledného PDF dokumentu.

