

1 MANUAL FOR THE TEMPLATE

The goal of this work was to revise and update an existing template created in 2017. The following sections provide a manual for the individual files, explaining how to work with them. The template provides a comprehensive format for dissertation and habilitation theses. It includes a structured layout of all essential parts, such as the main text, list of tables and figures, bibliography and author publications, and other appendices. It can be tailored to the needs of all UTB faculties and can set the main parts of the work to be written in either Czech or English. The template's primary aim is to ease the work of thesis authors, offering automated formatting to ensure key sections of the document are aligned with each faculty's standards. This chapter will sequentially introduce the various template sections and provide detailed instructions on how to use it for efficient thesis preparation. The individual files described here are not listed in the order they are used because their usage overlaps during the writing process.

1.1 Thesis.tex

The file thesis.tex forms the backbone of the template and contains the basic settings that have to be configured before starting the main text. First, you need to specify the type of thesis, determine the faculty for which it is intended, and choose the language in which it will be written. This is done using the following automated commands:

- **\SetType**

- **\SetFaculty**

- **\SetLanguage** In the next part of the file, the author fills in their personal details, as well as information that will appear on the title page, such as the title of thesis in Czech and English, regardless of the language in which the thesis is written. If the thesis is written in English, the order of the commands **\SetAbstractCZ** and **\SetAbstractEN** needs to be reversed so that the English abstract appears first.

The final section to be included in this file is a dedication or motto.

1.2 **Content.tex**

This file handles the the text, tables, images, and mathematical formulas that the author inserts, forming the main part of thesis. References or author publications are also included here. It features conditional command that allow the author to choose the language for chapters. The chapter titles can, of course, vary according to the needs of the author and the faculty. In such case, it is enough to change the description in the curly brackets following the **\section** command. The commands **\subsection** and **\subsubsection** are used for various sections of the document. If the thesis is extensive, which can occur in dissertation and habilitation theses, it is possible to reference other sections within a section:

```
1 \subsection {Subsection Title} \label{sec:intro>Title which
   will be used to reference the subsection}
```

Listing 1 Subsection

For a subsection set up this way, a reference within the thesis can be made using the command:

```
1 \ref{ Title which will be used to reference the subsection}
```

Listing 2 Subsection refernces

There is no need to manually number the chapters because the numbering is automatically set up in the template, and chapters will be numbered according to the order in which they were added. Every final thesis, regardless of the type, includes tables and images. **L^AT_EX** allows for various ways to work with these objects, and the following lines will describe how to use them and their options. The earliest I'll focus on the tables. Each table has to contain the essentials required for final theses because various sections of the document may refer to them and to ensure they are included in the “**List of Tables**”. These commands include **\caption** for naming the table and **\label** which allows the author to reference the

table in the document. Different $\text{\LaTeX}$ packages are used for working with tables, depending on the desired type. With the **tabularx** package, it is possible to create a table that adapts to a defined table width. The author can also set the width of specific columns in a proportion to the total table width. In the example below, I'll describe how we can set up such a table:

```

1 \begin{table}
2   \centering
3   \caption{Table Name}
4   \vspace{0.2cm}
5   \label{tab:table}
6   \begin{tabularx}{\textwidth}{|X|X|c|}
7     \hline
8     \textbf{Column Name 1} & \textbf{Column Name 2} &
9       1 & Lorem Ipsum is simply dummy text of the printing and
10      typesetting industry. & 123 \\ \hline
11      2 & Lorem Ipsum is simply dummy text of the printing and
12      typesetting industry. & 345 \\ \hline
13      3 & Lorem Ipsum is simply dummy text of the printing and
14      typesetting industry. & 789 \\ \hline
15   \end{tabularx}
16 \end{table}

```

Listing 3 Definition of the table environment

The line of code `\begin{tabularx}\textwidth|X|X|c|` starts the `tabularx` environment, which allows columns with the `X` parameter to automatically adapt to the specific width. All columns labeled with `X` have the same width unless specified otherwise. The content in a cell can be aligned in different ways. The parameter „c“ centers the content, „l“ aligns it to the left, and „r“ aligns it to the right. This rule generally applies to any table. Columns are separated by vertical lines, and if you write two vertical bars between the parameters, you will see two vertical lines between the columns. Horizontal lines are set up using the `\hline` command. The text between cells is written line by line, and the ampersand - `&`, separates

cells within a row. Another package that provides a specific table styles is **booktabs**. This package allows you to create a table without vertical borders and define various thickness for horizontal lines. To top border the table, use `\toprule`, `\midrule` for horizontal lines inside the table, and `\bottomrule` for the bottom line. To set a specific thickness for a horizontal line, simply write the desired pixel value in square brackets after the command defining the horizontal line. An example of a table using the booktabs package:

```

1 \begin{table}
2   \captionsetup{width=0.65\linewidth}
3   \caption{Table description 3}
4   \vspace{0.2cm} % Empty line
5   \label{tab:TableExample3}
6   \centering
7   \begin{tabular}{lcccccr}
8     \toprule[1px]
9     & 1 & 2 & 3 & 4 & 5 & Cena [Kč] & \\
10    \midrule
11    \emph{F} & (one) & (two) & (three) & (four) & (f) & 300
12    \\
13    \bottomrule[2px]
14  \end{tabular}
15 \end{table}

```

Listing 4 Definition of the table environment using the booktabs package

Besides using different types of tables, the author may need to highlight specific rows or cells in the table with colors. This feature is provided by the `[table]xcolor` package. To undercolor a specific cell, simply insert the `\cellcolor{Color}` command before the cell content. A list of colors supported by $\text{\LaTeX}$ can be found on Overleaf's website in the "Using Colors in $\text{\LaTeX}$ " document. Some colors might be too saturated, making them unsuitable for printing. To increase transparency, the author can add an exclamation mark "`\textbf{!}`" and the transparency percentage as a numeric value after the color name. This makes the color lighter and the text more readable.

```
\cellcolor{blue!25}300
```

Similarly, to undercolor a specific row in the table, the author can use the **\rowcolorColor** command before starting the **\begin{tabular}** environment. Transparency settings work the same way as in the previous example, where you can add an exclamation mark “!” and a numeric value for transparency after the color name.

```
1 \rowcolors{2}{grey!10}{red!20}
2 \begin{tabular}{|l|c|c|c|c|c|r|}
3   \hline
4 \end{tabular}
```

Listing 5 Setting row colors and beginning of the table

The **\rowcolors2grey!10red!20** command is used for alternating row colors in the table. The number 2 following the **\rowcolors** command specifies the starting row for undercolor. It means the first row of the table remains without undercolor and undercolor starts from the second row. Colors are set in the second and third parts of the command. The first color (grey!10) is used for even rows, while the second color (red!20) is for odd rows. This mechanism visually separates table rows for better readability. In $\text{\LaTeX}$ it is possible to create a table without adding any additional packages. In this case, Overleaf provides an automatic template. When the author uses the **\begin{table}** command and confirms it, the environment automatically completes the template, and the user only needs to fill data in the cells. Besides tables, the author will also need to work with images. If the user wants to insert images into the project, they first need to upload them to the folder located within the template directory. There is a folder named **graphics** specifically for this purpose. Just like tables, images must also contain the necessary commands to meet the standards for final papers. The **\caption** command is used to describe the image and **\label** provides a reference to the image within the document. The **graphicx** package is used for image handling. The author can then use a command to display the image in the document. Similar to tables, Overleaf provides an automatic template after typing **\begin{figure}** and confirming the command. Inside this command, the **\includegraphics[]** command defines the image path. The square brackets contain a width definition for the image relative to the page width, while curly brackets specify the path to the image in the

graphics folder.

```
\includegraphics[width=0.3\linewidth]{graphics/logo/fai_logo_cz.png}
```

If the author needs to insert multiple images side by side, simply use the appropriate number of `\includegraphics` commands and separate them with the “`\textbf`” symbol. Keep in mind that the width ratio of individual images should be adjusted accordingly. The `content.tex` file also contains examples of different bullet points that can be used in a LaTeX document. Bullet points can be created using the `\enumerate` and `\itemize` environments. The `\enumerate` environment generates numbered bullets for each point. Inside this environment, the `\item` command is used, followed by the point text. Each point is automatically numbered. The `\itemize` environment also uses the `\item` command, but unlike the previous example, each point has bullet points. If the author wants a specific character at the beginning of each point, it can be defined in square brackets following the `\item` command.

```
1 \begin{itemize}
2   \item[\textasteriskcentered] Lorem Ipsum with a star
3   \item[\$\bullet\$] Lorem Ipsum with a dot
4   \item[-] Lorem Ipsum with a dash
5 \end{itemize}
```

Listing 6 Definition of the `\itemize` environment

Here’s an example of how to use a star, dot, or dash as bullet points. The content should also include the literary sources the author worked with or their own publications referenced. Due to changes in the `UTB.tex` and `csplainnat.bst` files, which define the citation standard ČSN ISO 690:2022, the citation style has also changed. If sources from multiple files are used, like in this case, with `publications.bib` and `references.bib`, the correct source file must be specified in the work text. In `thesis.tex`, before the `\begindocument` command, I had to set parameters for new citations using this command:

```
1 \ifczech
```

```

2 \newcites{pub}{SEZNAM PUBLIKACÍ AUTORA}
3 \else
4 \newcites{pub}{PUBLICATIONS OF THE AUTHOR}
5 \fi

```

Listing 7 Working with the bibliography

Using the "pub" parameters, we can apply citations. To reference a source from the literature, use the command `\citeCitation` alias, and for the author's publications, use `\citepubCitation` alias.

1.3 Curriculum Vitae

In the template files, there is a file called `cv.tex`, which provides the author with a predefined CV template. This template was designed to simplify the author's work and create a unified visual style for both dissertation and habilitation thesis. The template is a suggested layout and can be modified by the author as needed, but it contains all the essential sections necessary for a CV. These include sections for personal information, education, work experience, other skills, and publication activities. If the author has created their own resume, they can use their own template. They should upload their resume in PDF format to the graphics folder and adjust the path to their resume in the `cv_import.tex` file.

1.4 pub.bib, references.bib

The author of a doctoral or habilitation thesis must cite the sources from which information was drawn, as well as their own publication activity. For this purpose, the template includes the files `pub.bib` and `references.bib`. The `pub.bib` file is used for the author's own publications, while `references.bib` handles literature sources. Each of these files works with a bibliography style definition: `csplainnat.bst` is used for `references.bib`, and `csplainnat_pub.bst` is used for the author's own publications. To add a source to a file, use the "@" symbol and choose the appropriate type based on whether it's a book, article, or online source. Overleaf automatically generate a template where the author fills in the relevant information. If additional source data needs to be included, add it within the command's curly brackets. Before adding

a particular field, verify that it's defined in `csplainnat.bst`. At the beginning of this file, there is a list of acceptable input data. If the desired parameter is missing, it is possible to use only the `note` command. If the author can't find the right source type via the "@"symbol, they can use the `misc{ }` command and fill in individual parameters for the source.