Institutions require a cover page and an affirmation at the end of the thesis. This package provides both. It stems from the computer science institutes of the University of Stuttgart, but is open to any university.

In case you are looking for a **full thesis template** including a cover page and affirmation, please go to https://latextemplates.github.io/scientific-thesis-template/. Please report issues concerning this package at the GitHub repository at https://github.com/latextemplates/scientific-thesis-cover.

Features

- utf8
- options for all required text on the coverpage

Installation

Usually, one does not need a seperate installation step, because both MiKTeX and texlive come with a recent version of this template. Do not forget to keep your tex distribution updated.

Alternatively, you can download `scientific-thesis-cover.sty` and put it in the folder where you are going to use it.

Usage Example

An example can be found in demo.tex.

Usage

Just include the package with all options specified:

```
\usepackage[
    title={Super relevant evaluation of new blackhole-generation method},
    author={Max Musterjunge},
    type=bachelor,
    institute=iaas,
    studyprogram=cs,
    examiner={Prof.\ Dr.\ Hans Mustermann},
    advisor={Otto Normalverbraucher, M.Sc.},
    startdate={2012-06-01},
    enddate={2012-12-01},
    language=english,
    logo={unilogo}
]{scientific-thesis-cover}
```

Afterwards you can create the cover using `\Coverpage` and get the affirmation text by using `\Affirmation`

Supported Options

This package supports the following options:

- language: Language used for all labels and text.
  - `language=german` will use german (default)
  - `language=english` will use english
- title: Title of work. Should be placed in curly braces:
  - `title={My thesis title}`
  - `title={My very long thesis title}`
- author: Author of work. Should be placed in curly braces. May contain more than one author seperated by commas:
  - `author={Peter Lustig}`
  - `author={Peter Lustig, Franz Josef, Vladimir Sixth}`
- type: Type of work. May be set to one of the following values or arbitrary text in curly braces:
  - `type=bachelor` will label your work as Bachelor's Thesis. Currently, the term regulations of the University of Stuttgart are in place. It is assumed that you study a German Bachelor program. Thus, even if you write in English, the type of your thesis is a "Bachelorarbeit". In case you study in a different program (such as INFOTECH), please fill in the appropriate type of your thesis in curly brackets. See below.
  - `type=master` will label your work as Masters's Thesis
  - `type=diplom` will label your work as Diploma Thesis
  - `type=study` will label your work as Student Research Project
  - `type=projectinf` will label your work as Projekt-INF
  - `type=prozessanalyse` will label your work as Prozessanalyse
  - `type=forschungsprojekt` will label your work as Bachelor-Forschungsprojekt Informatik
  - `type=projektaids` will label your work as Projekt Artificial Intelligence and Data Science
  - `type=studienprojektmi` will label your work as Studienprojekt Medieninformatik
  - `type=studienprojektse` will label your work as Studienprojekt Software Engineering
  - Arbitrary strings are also possible: `type={research project}` will label your work as "research project"
- institute: States for which institute you are doing this work. May be set to one of the following values or arbitrary text in curly braces:
  - `institute=iaas` will state Institute of Architecture of Application Systems

  - `institute=ipvs` will state Institute for Parallel and Distributed Systems
  - `institute=ipvsas` will state Institute for Parallel and Distributed Systems with department Application of Parallel and Distributed Systems (German: Anwendersoftware)
  - `institute=fmi` will state Institute of Formal Methods in Computer Science
  - `institute=ims` will state Institute for Natural Language Processing
  - `institute=iste` will state Institute of Software Engineering
  - `institute=istesqa` will state Institute of Software Engineering with department Software Quality and Architecture
  - `institute=iti` will state Institute of Computer Architecture and Computer Engineering
  - `institute=iris` will state Institute of Computer-aided Product Development Systems
  - `institute=vis` will state Institute of Visualization and Interactive Systems
  - `institute=visus` will state Visualization Research Center (VISUS)
  - `institute=sec` will state Institute of Information Security
  - `institute=fac` will state Faculty of Computer Science
  - Arbitrary strings are possible: `institute={Custom fictional institute}` will state Custom fictional institute
  - Several lines, e.g., for a department: `institute={IPVS\\Application of Parallel and Distributed Systems}` (see Multi-line values)
- department: Department of the institute, printed on its own line below the institute. Arbitrary text in curly braces; overrides the department of `istesqa` and `ipvsas`:
  - `department={Application of Parallel and Distributed Systems}`
- studyprogram: Your study program. The old name `course` still works. May be set to one of the following values or arbitrary text in curly braces:
  - `studyprogram=cs` will state that your study program is Computer Science
  - `studyprogram=se` will state that your study program is Software Engineering
  - `studyprogram=mcl` will state that your study program is Master Computational Linguistics
  - `studyprogram=msv` will state that your study program is Maschinelle Sprachverarbeitung
  - `studyprogram=bis` will state that your study program is Business Information Systems
  - `studyprogram=simtech` will state that your study program is Simulation Technology

  - `studyprogram=medinf` will state that your study program is Medieninformatik
  - `studyprogram=aids` will state that your study program is Artificial Intelligence and Data Science
  - `studyprogram=swe` will state that your study program is Software Engineering (the Bachelor programme since 2025; `se` gives the older name Softwaretechnik)
  - Arbitrary strings are possible: `studyprogram={New Study program}` will state that your study program is New Study program

- examiner: Your examiner.
  - `examiner={Prof.\ Dr.\ Hans Mustermann}`
- advisor: Your advisor. The old name `supervisor` still works.
  - `advisor={Otto Normalverbraucher, M.Sc.}`
- startdate: Startdate of your work. Preferably ISO-8601. See https://xkcd.com/1179/ and https://www.explainxkcd.com/wiki/index.php/1179:_ISO_8601.
  - `startdate={2012-06-01}`
- enddate: Enddate of your work.
  - `enddate={2012-12-01}`
- crk: CR-Classification codes of your work. May be separated by commas:
  - `crk={A.1, A.2}`

Additional Optional Options

- number: Running number of work. May contain arbitrary text. Should contain the number you got for your work.
  - `number=1234` will label your work to have number 1234
- logo: Adds a logo, e.g. of your university, at the top of the coverpage.
  - The image's path can not contain any underscores!
  - Without a logo, the name of the university is printed in its place.
  - The logo of the University of Stuttgart is not bundled (trademark).
- `setPageNumberToOne=true` will set the page after the cover to `1` (default false)
- `setCoverPageNumberToMinusOne=true` will set `-1` as the page number for the cover page (default false)

Multi-line values

Option values may contain `\\` and commands, e.g., to state a department below the institute or to name two advisors:

```
institute={IPVS\\Application of Parallel and Distributed Systems},
advisor={Otto Normalverbraucher,\ M.Sc.\\Prof.\ Dr.\ Black},
```

Each block is centered (institute) or left-aligned (examination data) line by line. With an older LaTeX distribution, load `kvoptions-patch` **before** the document class to keep `\\` in option values:

```
\RequirePackage{kvoptions-patch}
\documentclass[twoside]{article}
```

How to release

1. Check if indent is correct: `latexindent -y="indentPreamble:1,defaultIndent:'  '" -m -w scientific-thesis-cover.sty`
2. Adapt `CHANGELOG.md`
3. Adapt version and date in `scientific-thesis-cover.sty`
4. Tag the release commit using `git tag`
5. Push to GitHub
6. Add CHANGELOG part to GitHub release manually, because github-release-from-changelog currently works with `package.json` only.
7. Publish to CTAN

- Pre-conditions:
  - Windows: Patch `C:\MiKTeX\scripts\ctanify\ctanify`: Remove `-y` from `zip` command line arguments `zip -q -r -9 -y -m`.
  - Windows: Environment variable `TMP` must not contain backslashes. E.g., `C:/TEMP/WIN` is good, `C:\Users\user\AppData\Local\Temp` is bad.
  - Have `pandoc` and `sed` in the path.
- Run `release.bat`.
- Go to https://ctan.org/pkg/scientific-thesis-cover, choose "Upload"
- Use `scientific-thesis-cover.tar.gz` as archive.

7. Adapt `CHANGELOG.md` to contain `## [unreleased]` again.

License

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, version 1.3c of the license. The latest version of this license is in http://www.latex-project.org/lppl.txt and version 1.3c or later is part of all distributions of LaTeX version 2005/12/01 or later.

This work has the LPPL maintenance status "maintained".

The Current Maintainer and author of this work is Oliver Kopp.

This work consists of the files `scientific-thesis-cover.sty`, `demo.tex`, and `README.md`. In case files are not listed here, but available at https://github.com/latextemplates/scientific-thesis-cover, these additional files also form part of this work.

Contributors (incomplete list)

Bernd Raichle, Timo Heiber, Steffen Keul, Oliver Kopp, Kai Mindermann, Matthias Papesch, Nils Radtke, Niklas Schnelle, Tim Halach