The ufrgscca, and associated, Packages
Version 1.0.1

Alceu Frigeri

September 2022

Abstract

This bundled is aimed at producing undergraduate students final work/report at UFRGS/EE (Engineering School at the Federal University of Rio Grande do Sul), closely following ABNT rules (Brazilian Association for Technical Norms). It is composed of a main class, `ufrgscca`, and a set of auxiliary packages, some of which can be used independently.

Contents

1 Introduction . . . 1
1.1 Current Version . . . 2
2 ufrgscca Class . . . 2
2.1 Class Options . . . 2
2.2 Class Declared Commands . . . 3
3 ufrgscca-abnt Package . . . 3
3.1 Package Options . . . 4
3.2 Environments . . . 5
3.3 Tabular New Columns . . . 6
3.4 enumitem Extra Keys . . . 6
4 ufrgscca-core Package . . . 8
4.1 Core Forms Commands . . . 8
4.2 Core Global Commands . . . 9
4.3 Core Specific Commands . . . 9
5 ufrgscca-cover Package . . . 9
5.1 Package Options . . . 9
5.2 Defined Commands . . . 9
6 ufrgscca-forms Package . . . 10
6.1 Forms Defined Commands . . . 10
7 ufrgscca-lists Package . . . 10
7.1 Environment . . . 10
7.2 Declared Commands . . . 11

1 Introduction

ABNT rules can be quite challenging some times (read: bibliography style/references) and sometimes just odd (line spacing, front matter, page layout), nevertheless it is a *Brazilian Standard* for typography whose students at UFRGS should grow cherished to follow.

In short, as of version 1.0.1 the bundle is composed of a class, *ufrgscca* (based on the standard LaTeX2e *report* class), which pre-loads all other, as needed, packages: *ufrgscca-abnt*, *ufrgscca-core*, *ufrgscca-cover*, *ufrgscca-forms*, *ufrgscca-gen*, *ufrgscca-lists*, *ufrgscca-curr*, *ufrgscca-coord*, *ufrgscca-ppc*.
N.B.: This bundle requires a quite recent LaTeX2e kernel, at least as recent as June 2022, which allows to declare package options using the new *key* = *value* system and declare commands with `\NewDocumentCommand`, out-of-the-box.

1.1 Current Version

For the sake of the 'maintainers' sanity, since this is a bundle, all files are saved with the same version (bundle version), with two exceptions: *ufrgscca-curr.sty* *ufrgscca-ppc.sty* which are less tested than the others, and somewhat in what one would call 'beta' state. Better said, all files are version 1.0.1, except *ufrgscca-curr* and *ufrgscca-ppc* whose versions are 1.0.1beta.

2 ufrgscca Class

The following packages are always pre-loaded: *etex, etoolbox, lmodern, fontenc (T1), inputenc (utf8), silence, ufrgscca-abnt, ufrgscca-gen, ufrgscca-cover, ufrgscca-core, hyperref* and (if it exists) a *local.tex* file.
Other set of auxiliary packages are also pre-loaded, depending on the class options used, and finally it loads (normally) the *report* class (the exception being if one uses the `dctools` option).
Being based on the report class, one can use all class options one would with a report, plus the ones listed below.

2.1 Class Options

tocdepth use: *tocdepth* = ⟨number⟩, whereas ⟨number⟩ indicates the deepest sectioning to appears in the Table of Contents (0 being the top section, which is `\chapter` for report based classes, 1 being `\section`, and so on.) The default value being 3 (`\subsubsection`).

secdepth use: *secdepth* = ⟨number⟩, whereas ⟨number⟩ indicates the deepest sectioning to be numbered. (0 being the top section, which is `\chapter` for report based classes, 1 being `\section`, and so on.) The default value being 4 (`\paragraph`).

english the default language being Portuguese, this option changes locale to English.

brazilian in some rare cases (to be further investigated) babel seems to get confused about which language is active, this "shouldn't be necessary" but one can explicitly tell babel to use THIS language (which should, otherwise, be the default one).

relnum by default, figures, tables, etc. are numbered as a continuous series. With this switch, those lists are reset at each chapter, e.g. Figure 5.1 instead of Figure 23.

openright in case of printed material, this will assure that a `\chapter` always starts at an odd page, which is relevant in case of printing out (double sided) the document.

oneside in case the document will be printed in single side sheets, otherwise it's assumed a two-sided printing.

strict-abnt to assure asymmetric margins, as defined by ABNT: inner ones greater than outer ones, which matters if you are going to print the doc and make a book of it, but makes it odd to look at in a computer screen, reason by which the current default setting is for symmetric margins (same text width).

`repeatfields` in case of authors with multiple publications, their names will be repeated for each entry. In the default setting the author's name is written only in the first entry, and replaced by underscores in the other entries.

`xlists` this will load the ufrgscca-lists package, for the definition of new floats/lists.

`xpacks` this will load a series of packages, which can be handy when writing Engineering reports: *relsize, keyval, graphicx, mathtools, mathrsfs, amsfonts, amssymb, empheq, amsthm, extarrows, mathfixs, bigdelim, circuitikz, steimenz* and tikz libraries: *fit, math, calc, shapes.geometry, shapes.misc, shapes.multipart, graphs, 3d, positioning, shadows, babel*. One is advised to look after each package documentation (ctan.org) for further information.

`report` in case the doc is just a class assignment with, possibly, many co-authors. It changes mainly the front matter, which is simplified (no referral page, for instance).

`internship` in case the doc is an internship report.

`forms` in the process of submitting a student final work/report, there is a series of forms to be submitted, this allows the customization of said forms in a simple way.

`chapternopagenum` to suppress the page numbers at chapters begin.

`nomicrotype` in some rare cases, *`microtype`* might hurt page layout, this allows the suppression of *`microtype`*.

`showframes` for layout proof only, it will draw frames around each page main parts.

`showlabels` it will put a reference mark in each label created, and print out it's name.

`nofontwarning` in case of *`ufrgscca-ppc`* is loaded, it will suppress some font related warnings.

`dctools` this will change page layout and base class to article, it is meant to document the class itself.

`texlive` this is a reserved key, in case some workaround for texlive is needed.

`overleaf` this is a reserved key, in case some workaround for overleaf is needed.

`miktex` this is a reserved key, in case some workaround for miktex is needed.

2.2 Class Declared Commands

`\autonameref`
`\annexref`
`\autoannexref`

```
\autonameref [⟨sep⟩] {⟨label⟩} [⟨spc⟩]
\annexref {⟨label⟩}
\autoannexref [⟨sep⟩] {⟨label⟩} [⟨spc⟩]
```

The *`hyperref`* package, sometimes, gets the `\autoref` name wrong (when referencing an annex), therefore the `\annexref {⟨label⟩}` will assure the correct annex name is used.

`\autonameref {⟨label⟩}` produces an entry of the form '`\autoref {⟨label⟩} ⟨sep⟩ \nameref {⟨label⟩} ⟨spc⟩`'

`\autoannexref {⟨label⟩}` produces an entry of the form '`\annexref {⟨label⟩} ⟨sep⟩ \nameref {⟨label⟩} ⟨spc⟩`'

The default ⟨sep⟩ being a comma, and the default ⟨spc⟩ being empty space.

3 ufrgscca-abnt Package

This package is the one that sets the page layout (using *`geometry`*, *`titlesec`*, *`titletoc`*) and adjusts the main float environments (figure, tables, captions). It can be used as a stand alone package, regardless of the underlying class.

The following packages are always pre-loaded: *babel, csquotes, geometry, appendix, titlesec, titletoc, enumitem, chngctr, caption, biblatex, microtype, array, nicematrix, contour* and *soul*.

Take note that *biblatex* is loaded with the **biber** option, to correctly handle ABNT biography style.

3.1 Package Options

strict-abnt to assure asymmetric margins, as defined by ABNT: inner ones greater than outer ones, which matters if you are going to print the doc and make a book of it, but makes it odd to look at in a computer screen, reason by which the current default setting is for symmetric margins (same text width).

chapternopagenum to suppress the page numbers at chapters begin.

relnum by default, figures, tables, etc. are numbered as a continuous series. With this switch, those lists are reset at each chapter, e.g. Figure 5.1 instead of Figure 23.

repeatfields in case of authors with multiple publications, their names will be repeated for each entry. In the default setting the author's name is written only in the first entry, and replaced by underscores in the other entries.

nomicrotype in some rare cases, *microtype* might hurt page layout, this allows the suppression of *microtype*.

showframes for layout proof only, it will draw frames around each page main parts.

showlabels it will put a reference mark in each label created, and print out it's name.

tocdepth use: *tocdepth* = ⟨number⟩, whereas ⟨number⟩ indicates the deepest sectioning to appears in the Table of Contents (0 being the top section, which is `\chapter` for report based classes, 1 being `\section`, and so on.) The default value being 3 (`\subsubsection`).

secdepth use: *secdepth* = ⟨number⟩, whereas ⟨number⟩ indicates the deepest sectioning to be numbered. (0 being the top section, which is `\chapter` for report based classes, 1 being `\section`, and so on.) The default value being 4 (`\paragraph`).

dctools this will change page layout and base class to article, it is meant to document the class itself.

`\keyword`

`\keyword{⟨keyword⟩}`

This command can be invoked many times, it will construct a list of keywords to be used when printing out the abstract environment.

`\sourcecitation`
`\note`

`\sourcecitation{⟨source⟩}`
`\note{⟨text⟩}`

When describing floating elements (like figure, tables, circuits) one always has to cite the source of it, and in some cases it might be necessary to add a special note. Those assure uniformity when doing that.

`\nonum`
`\notoc`

`\nonum\chapter{⟨chap.title⟩}`
`\nonum\section{⟨sec.title⟩}`
`\notoc\chapter{⟨chap.title⟩}`
`\notoc\section{⟨sec.title⟩}`

In some cases, it might be necessary to create a numberless chapters or sections. Those two commands can be used as a *prefix* to any sectioning command. Whilst `\nonum` will just suppress the sectioning number, the `\notoc` will also suppress it from the table of contents.

LaTeX Code:

```
\nonum\chapter{some title} %this one will appear in the toc
\notoc\section{some other title} %this won't even appear in the toc
```

`\tightul`

`\tightul{⟨text⟩}`

This will *underline* a short text, take note that ⟨text⟩ 'can't be broken' (think paragraph justification), which can lead to *text overflows* and bad justification.

LaTeX Code:	LaTeX Result:
`\tightul{Some text example}%`	Some text example

\NewChapListEnv

\NewChapListEnv{⟨envname⟩}{⟨displayname⟩}

This is the command used to created those *chapter like* lists, like 'List of Symbols' or 'List of acronyms'. With it, a new environment is created, ⟨envname⟩, with an associated 'numberless' chapter name ⟨displayname⟩. The newly created environment will implement a *description* like environment (thanks to *enumitem*) with an optional and a mandatory argument (see below).

LaTeX Code:

```
\def\listabbrvname{Lista de Abreviaturas}
\NewChapListEnv{listofabbrv}{\listabbrvname} % this is the actual code
    used in ufrgscca-abnt.sty
```

\date
\today
\monthname

\date[⟨day⟩]{⟨month⟩}{⟨year⟩}
\today
\monthname

The command \date is redefined, to allow a separation between the many arguments ⟨day⟩, ⟨month⟩ and ⟨year⟩. If not called by the user it *defaults* to current month / year. \today returns the current *locale* date, whilst \monthname returns the *locale* name of the current month.

3.2 Environments

abstract

\begin{abstract}[⟨lang⟩]{⟨keywords⟩}...\end{abstract}

The standard environment *abstract* is redefined as a numberless chapter based on the current locale (default: Portuguese), at the end of it the keywords list created with \keyword will be added.

LaTeX Code:

```
\keyword{a keyword}
\keyword{another keyword}
\begin{abstract} some short summary of things\ldots
\end{abstract}
```

otherabstract

\begin{otherabstract}[⟨lang⟩]{⟨keywords⟩}...\end{otherabstract}

This is the environment to create an abstract in a language other than the default one. The default value for ⟨lang⟩ is english, and it can be any value that *babel* understands. The ⟨keywords⟩ are just a list of keywords which will be added at the end of the *otherabstract*.

LaTeX Code:

```
\begin{otherabstract}[english]{a keyword, another keyword} some short
    summary of things\ldots
\end{otherabstract}
```

listofabbrv
listofsymbols

\begin{listofabbrv}[⟨enum-opt⟩]{⟨ABBRV⟩}...\end{listofabbrv}
\begin{listofsymbols}[⟨enum-opt⟩]{⟨SYMB⟩}...\end{listofsymbols}

Both environments create a description like list preceded by a numberless (\nonum) chapter. ⟨enum-opt⟩ is any *enumitem* list valid key. Whereas ⟨ABBRV⟩ / ⟨SYMB⟩ are just the 'biggest' abbreviation/symbol to be used as a tab reference.

appendix
annex

\begin{appendix}.... \end{appendix}

\begin{annex}.... \end{annex}
Those two environments start the appendices and annex chapters (using locale). Chapters are alphabetic *numbered* (starting at A).

3.3 Tabular New Columns

Thanks to array some new columns types are defined:

P P{⟨width⟩} Normal text, ragged left.
B B{⟨width⟩} Bold text, ragged left.
C C{⟨width⟩} Normal text, centered.
R R{⟨width⟩} Normal text, ragged left.
L L{⟨width⟩} Normal text, ragged right.
J J{⟨width⟩} Normal text, justified.

3.4 enumitem Extra Keys

Besides the *default* keys defined by the enumitem package a few others are defined for author's convenience:

ppc, tcc ppc and tcc are alias of each other, and just assure that lists indentation will be the same as paragraphs default.
parindent with parindent, the list number/mark is aligned with paragraph indentation.
noindent noindent removes the label indentation.

LaTeX Code: LaTeX Result:

```
\begin{enumerate}[tcc]
  \item some A
  \item some B
\end{enumerate}
\begin{enumerate}[tcc,parindent]
  \item some A
  \item some B
\end{enumerate}
\begin{enumerate}[parindent]
  \item some A
  \item some B
\end{enumerate}
\begin{enumerate}[noindent]
  \item some A
  \item some B
\end{enumerate}

New paragraph, for reference.
```

1. some A

2. some B

1. some A

2. some B

1.some A

2.some B

1.some A

2.some B

New paragraph, for reference.

tight allows for very tight lists (no indentation) to be used, for instance, inside quotes. N.B. don't use it in normal paragraph mode, otherwise the labels will spill outside the default text window.
miditemsep miditemsep halves items separation, as an alternative to noitemsep from enumitem

LaTeX Code: LaTeX Result:

```
\begin{enumerate}[tcc]
  \item some A
  \item some B
\end{enumerate}
\begin{enumerate}[tcc,miditemsep]
  \item some A
  \item some B
\end{enumerate}
\begin{enumerate}[tcc,noitemsep]
  \item some A
  \item some B
\end{enumerate}
```

1. some A

2. some B

1. some A
2. some B

1. some A
2. some B

`arabic` That's the *default* enumerate style. Arabic numbers, starting at 1, followed by a dot.

`arabic)` Label will be constructed as number followed by a parenthesis.

`(arabic)` Label will be enclosed by parenthesis.

`arabic*` (for secondary lists) Label will be constructed by the label of the outer list, this item number and a final dot.

`arabic*)` (for secondary lists) Label will be constructed by the label of the outer list, this item number and a final parenthesis.

`roman` This and below keys are the same as the arabic ones, but using lower case roman numbers.

`roman)` lower case roman number, followed by a parenthesis.

`(roman)` enclosed by parenthesis.

`roman*` preceding one followed by roman number and a final dot.

`roman*)` same, followed by a final parenthesis.

`Roman` This and below keys are the same as the arabic ones, but using upper case roman numbers.

`Roman)` upper case roman number, followed by a parenthesis.

`(Roman)` enclosed by parenthesis.

`Roman*` preceding one followed by roman number and a final dot.

`Roman*)` same, followed by a final parenthesis.

`alpha` This and below keys are the same as the arabic ones, but using lower case alpha numbers.

`alpha)` lower case alpha number, followed by a parenthesis.

`(alpha)` enclosed by parenthesis.

`alpha*` preceding one followed by alpha number and a final dot.

`alpha*)` same, followed by a final parenthesis.

`Alpha` This and below keys are the same as the arabic ones, but using upper case alpha numbers.

`Alpha)` upper case roman number, followed by a parenthesis.

`(Alpha)` enclosed by parenthesis.

`Alpha*` preceding one followed by roman number and a final dot.

`Alpha*)` same, followed by a final parenthesis.

LaTeX Code: LaTeX Result:

```
\begin{enumerate}[tcc,roman]
  \item some A
  \item some B
  \item some C
\end{enumerate}
\begin{enumerate}[tcc,Roman]
  \item some A
  \item some B
  \begin{enumerate}[tcc,alpha*]
    \item some A
    \item some B
    \item some C
  \end{enumerate}
  \item some C
\end{enumerate}
\begin{enumerate}[tcc,arabic]
  \item some A
  \item some B
  \begin{enumerate}[tcc,roman*)]
    \item some A
    \item some B
    \item some C
  \end{enumerate}
  \item some C
\end{enumerate}
```

i. some A

ii. some B

iii. some C

I. some A

II. some B

  II.a. some A
  II.b. some B
  II.c. some C

III. some C

1. some A

2. some B

  2.i) some A
  2.ii) some B
  2.iii) some C

3. some C

bullet for simple itemized lists, it will replace the default black dot by an ‘open bullet’

LaTeX Code: LaTeX Result:

```
\begin{itemize}[tcc,miditemsep]
  \item some A
  \item some B
  \item some C
\end{itemize}
\begin{itemize}[tcc,bullet,
    miditemsep]
  \item some A
  \item some B
  \item some C
\end{itemize}
```

- some A
- some B
- some C

◦ some A
◦ some B
◦ some C

4 ufrgscca-core Package

The *ufrgscca-core* package defines a set of commands for authors, students, advisors and examiners names and related info. It is needed by most/all of the tc bundled packages.

4.1 Core Forms Commands

\tccbrief
\tcccoadvisorbrief
\tccadvisorsreview

\tccbrief {⟨brief⟩}
\tcccoadvisorbrief {⟨brief⟩}
\tccadvisorsreview {⟨brief⟩}

Those commands are only of use when using *ufrgscca-forms*. \tccbrief sets the work initial summary, \tcccoadvisorbrief sets the justification for having a co-advisor, `\tccadvisorsreview` sets the advisor's review.

4.2 Core Global Commands

`\location`

```
\location{⟨city⟩}{⟨state⟩}
```

To redefine the default values of ⟨city⟩ and ⟨state⟩ (Porto Alegre and RS).

`\TCCcoord`
`\TCCcoordtitle`

```
\TCCcoord{⟨(title) full name⟩}[⟨gender⟩]
\TCCcoordtitle{⟨coordinator denomination⟩}
```

`\coursecoord`
`\coursecoordtitle`

```
\coursecoord[⟨(title) full name⟩][⟨gender⟩]
\coursecoordtitle{⟨course coordinator denomination⟩}
```

⟨coordinator denomination⟩ and ⟨course coordinator denomination⟩ are the full 'job title' of their position. ⟨gender⟩ can be either 'm' or 'f'.

4.3 Core Specific Commands

The following commands are more or less self-explanatory, ⟨ID⟩ is the student's university ID. ⟨Nproc⟩ is the process/request number. ⟨gender⟩ can be either 'm' or 'f'.

`\author`
`\authorinfo`
`\student`
`\studentinfo`

```
\author{⟨last⟩}{⟨first⟩}[⟨gender⟩]
\authorinfo[⟨Nproc⟩]{⟨ID⟩}{⟨email⟩}
\student{⟨last⟩}{⟨first⟩}[⟨gender⟩]
\studentinfo[⟨Nproc⟩]{⟨ID⟩}{⟨email⟩}
```

`\advisor`
`\advisorinfo`

```
\advisor[⟨title⟩]{⟨last⟩}{⟨first⟩}[⟨gender⟩]
\advisorinfo{⟨Institut⟩}{⟨title-info⟩}{⟨email⟩}{⟨phone⟩}
```

`\coadvisor`
`\coadvisorinfo`

```
\coadvisor[⟨title⟩]{⟨last⟩}{⟨first⟩}[⟨gender⟩]
\coadvisorinfo{⟨Institut⟩}{⟨title-info⟩}{⟨email⟩}{⟨phone⟩}
```

`\examiner`
`\examinerinfo`

```
\examiner[⟨title⟩]{⟨last⟩}{⟨first⟩}[⟨gender⟩]
\examinerinfo{⟨Institut⟩}{⟨title-info⟩}{⟨email⟩}{⟨phone⟩}
```

`\altexaminer`
`\altexaminerinfo`

```
\altexaminer[⟨title⟩]{⟨last⟩}{⟨first⟩}[⟨gender⟩]
\altexaminerinfo{⟨Institut⟩}{⟨title-info⟩}{⟨email⟩}{⟨phone⟩}
```

5 ufrgscca-cover Package

This package is the one that sets the front pages, depending on the kind of 'report' being generated. The default being to generate 3 cover pages: an identification on, followed by presentation one, then an referral/approval one.

5.1 Package Options

report in case the doc is just a class assignment with, possibly, many co-authors. It changes mainly the front matter, which is simplified (no referral page, for instance).

internship in case the report is a internship one.

5.2 Defined Commands

`\maketitle`

```
\maketitle
```

This is the only main command, which will typeset the front matter. It requires that all *specific info* be already set up (like work title, author's name, affiliation, etc.)

\course
\courseacronym
\graduationtitle
\university
\universityacronym
\universitydivision
\divisiongradcouncil
\department
\classcode
\classname
\subject

\course {⟨arg⟩}
\courseacronym {⟨arg⟩}
\graduationtitle {⟨arg⟩}
\university {⟨arg⟩}
\universityacronym {⟨arg⟩}
\universitydivision {⟨arg⟩}
\divisiongradcouncil {⟨arg⟩}
\department {⟨arg⟩}
\classcode {⟨arg⟩}
\classname {⟨arg⟩}
\subject {⟨arg⟩}
In case some customization is needed, one can change them as needed. The default values are set for the *control and automation* course at UFRGS/EE.

6 ufrgscca-forms Package

This package defines just two user commands to generate specific forms needed at UFRGS/EE.

6.1 Forms Defined Commands

\tcforms
\tcemptyforms

\tcforms {⟨formslist⟩}
\tcemptyforms {⟨formslist⟩}
\tcforms will generate the many forms (⟨formslist⟩) using the information from *local.tex*, whilst \tcemptyforms will generate said forms with 'blanks' (to be fulfilled by hand, for instance).
⟨formslist⟩ is a csv list of any of:

reqform	Registration requirement form.
coadvisor	Coadvisor justification form.
boardsapproval	Boards approval form.
advisorsapproval	Advisors approval form.
receipts	Receipts forms (one per board member).
examinersforms	Grades and correction forms (per board member).
rectifyapprovalform	Corrections approval form.

7 ufrgscca-lists Package

The following packages are always pre-loaded: *newfloat, listings* and *xcolor*. It defines a new *floating environment*. Combined with *listings* one can typeset exempts of *code listing*.

7.1 Environment

codelist

\begin{codelist}...\end{codelist}

\caption will be named ’Listing’ (Listagem).
LaTeX Code:

```
\begin{codelist}[htbp]
  \caption{sample C code}
  \label{code01}
  \begin{lstlisting}[language=C]
    struct i2c_msg
    {
      __u16 addr;     /* endereco do escravo */
      __u16 flags;
    }
  \end{lstlisting}
  {\sourcecitation{\textcite{Garg:SMA-2000}}}
\end{codelist}
```

7.2 Declared Commands

listofcodelist

\listofcodelist
This will create the ’List of ...’ associated with the previous environment.