891923 · Ramolivaz André
892614 · Tomasin Alberto
892539 · Dinato Simone

EUResearchHub: Progresso in sinergia

Report per il progetto di CT0006 - Basi di Dati

Ca' Foscari
University
of Venice

Indice

1 Introduzione **4**
1.1 Contesto dell'applicazione . . . 5
1.2 Obiettivi e scopo del documento . . . 5

2 Panoramica delle funzionalità del sistema **6**
2.1 Gestione dei progetti di ricerca . . . 7
2.2 Processo di valutazione e storico modifiche . . . 7
2.3 Interazione tra ricercatori e valutatori . . . 7

3 Progettazione e modellazione della base di dati **8**
3.1 Raccolta ed analisi dei requisiti . . . 9
3.1.1 Quadro generale . . . 9
3.1.2 Integrità dei dati: trigger, vincoli e transazioni . . . 9
3.1.3 Gestione dei ruoli e politiche di autorizzazione . . . 9
3.1.4 Performance: indici, materialized view e procedure . . . 9
3.2 Progettazione concettuale . . . 9
3.2.1 Schema ad oggetti . . . 9
3.2.2 Giustificazione delle decisioni progettuali . . . 9
3.3 Progettazione logica . . . 9
3.3.1 Schema logico . . . 9
3.3.2 Giustificazione delle decisioni progettuali . . . 9
3.4 Progettazione fisica . . . 9
3.4.1 Istruzioni `SQL` . . . 9

4 Implementazione back-end **10**
4.1 Flask e SQLAlchemy . . . 11
4.2 Expression Language e ORM . . . 11
4.3 Sicurezza . . . 11
4.3.1 Protezione da XSS . . . 11
4.3.2 Protezione da SQL Injection . . . 11
4.3.3 Meccanismi di sicurezza aggiuntivi . . . 11
4.4 Approfondimento delle tecnologiche specifiche . . . 11
4.4.1 `libreria1` . . . 11
4.4.2 `libreria2` . . . 11

5 Implementazione front-end **12**
5.1 Brand identity . . . 13
5.2 Demo . . . 13
5.3 Approfondimento delle tecnologiche specifiche . . . 13
5.3.1 `framework1` . . . 13
5.3.2 `css` . . . 13
5.3.3 `javascript` . . . 13

6 Collaborazione e contributo individuale **14**
6.1 Planning . . . 15
6.2 Descrizione del ruolo di ciascun membro del gruppo . . . 15

Capitolo 1

Introduzione

1.1 Contesto dell'applicazione

The `imta_core` package provides a LaTeX template that satisfies the IMT Atlantique corporate identity rules. [1] For slightly more advanced or specific features, you might want to refer to the `imta_extra` package.

Since it is a package, it can be used for a variety of classes and geometry. It has been primarily designed to be used in `article` and `report` documents, either with `oneside`/`twoside` or `onecolumn`/`twocolumn` geometries. If you are not sure how to start, we invite you to have a look at the `skeleton` file provided with this template.

1.2 Obiettivi e scopo del documento

In order to use this template, you simply need to have a TeX distribution installed, copy this file in your working directory and add the `\usepackage {imta_core}` command in the preamble of your top document. The `babel` package should be loaded first when used in combination with `imta_core`.

If you would prefer to properly install the package once it for all instead of having multiple copies, you can do so by manually installing it. Since this can be a little bit tedious for a new LaTeX user, we introduced a custom Python script. Simply run the `imta_install.py` script and everything should be alright. This script supports Python 2 and 3, and both MikTeX and TeXLive environments.

Note that this template is compatible with at least TeXLive and MiKTeXdistributions, although we have noticed some issues might happen when trying to resolve dependencies with the latter. If so, then try to update your distribution using your MikTeXmanager. [2]

[1] See *Charte Graphique* on the IMT Atlantique's intranet for more details.
[2] See *General Guide to Installing Packages with MikTeX Package Manager* on TeXStack Exchange for more details.

Capitolo 2

Panoramica delle funzionalità del sistema

2.1 Gestione dei progetti di ricerca

2.2 Processo di valutazione e storico modifiche

2.3 Interazione tra ricercatori e valutatori

Capitolo 3

Progettazione e modellazione della base di dati

3.1 Raccolta ed analisi dei requisiti

3.1.1 Quadro generale

3.1.2 Integrità dei dati: trigger, vincoli e transazioni

3.1.3 Gestione dei ruoli e politiche di autorizzazione

3.1.4 Performance: indici, materialized view e procedure

3.2 Progettazione concettuale

3.2.1 Schema ad oggetti

3.2.2 Giustificazione delle decisioni progettuali

3.3 Progettazione logica

3.3.1 Schema logico

3.3.2 Giustificazione delle decisioni progettuali

3.4 Progettazione fisica

3.4.1 Istruzioni `SQL`

Capitolo 4

Implementazione back-end

4.1 Flask e SQLAlchemy

4.2 Expression Language e ORM

4.3 Sicurezza

4.3.1 Protezione da XSS

4.3.2 Protezione da SQL Injection

4.3.3 Meccanismi di sicurezza aggiuntivi

4.4 Approfondimento delle tecnologiche specifiche

4.4.1 `libreria1`

4.4.2 `libreria2`

Capitolo 5

Implementazione front-end

5.1 Brand identity

5.2 Demo

5.3 Approfondimento delle tecnologiche specifiche

5.3.1 framework1

5.3.2 css

5.3.3 javascript

Capitolo 6

Collaborazione e contributo individuale

6.1 Planning

6.2 Descrizione del ruolo di ciascun membro del gruppo

The `\imtaQuestion` command outputs and formats a question counter. It's meant to be used in reports for assignment with questions. The counter should be reset with the `\imtaQuestionReset`. This couple of commands is meant to be used for sectioning when the assignment does not use a more explicit titling.

The `imta_core` package offers a one-level-deeper section than the usual deepest `\subsubsection`. This provides an alternative to the usual `\paragraph`.

The `\chapter` command has been redefined to print the word "chapter" in front of the figure. This command is compatible with several languages[1] but requires the `babel` package to be loaded first. The default language is english.

This command also prints a small design on the bottom right corner of the page and updates the upper section title. In case the document is in `twoside` mode, it ensures the chapter page is on the right by inserting a blank page if necessary. In case the document is in `twocolumn` mode, it also temporarily restores a `onecolumn` geometry to print the chapter page.

The core package defines four colors, including the three colors of the IMT Atlantique, and a uniform and arbitrary gray. These are defined as follows:

```
\definecolor{imtaGreen}{RGB}{164, 210, 51}
\definecolor{imtaLightBlue}{RGB}{0, 184, 222}
\definecolor{imtaDarkBlue}{RGB}{12, 35, 64}
\definecolor{imtaGray}{RGB}{87, 87, 87}
```

Here are samples of these colors, with text in both black and white for previsualising the contrast.

Figura 6.1: Samples of the IMT Atlantique colors

The official IMT Atlantique styling is not really LaTeX-ish, and takes the decision to use a sans-serif font for body text. Therefore, the default style uses the default LaTeX font settings. However, it is possible to enable a more IMT Atlantique-compliant styling, by calling the `\imtaSetIMTStyle` command in the preamble.

The main aspects of the official style are:

[1] This functionality currently supports english, french, german, portuguese and spanish.

- Use of the Helvetica font for the body;
- Section titles in green (`\imtaGreen`) and other heading titles in gray (`\imtaGray`);
- Section title in the header;
- Page number at the right corner of the footer.

For comparison, the default style of the template is:

- Use of the default Computer Modern font for the body;
- Default style for headings: all in black;
- Document title at the left corner and author's name at the right corner of the header;
- Page number at the center of the footer.

The IMT Atlantique logo can be output at the desired width with the `\imtaLogo` command. The latter includes an external pdf document, `imta_logo.pdf`, that contains the official logo. On the other hand, the `\imtaLogoTikz` draws an approximation of the logo with the `tikz` package. The following is a comparison of both commands (with an added frame). Note that the current version of this template uses `\imtaLogoTikz` for both the front and back cover.

Figura 6.2: Comparison between `imtaLogo` (left) and `imtaLogo` (right)

The `imtaMaketitlepage` command outputs a title page with the names of the authors, the date of writing, and the title of the document, along with the subtitle. For this latter purpose, the `\subtitle` command helps define a subtitle as a part of the document's metadata, and is used inside of the `\imtaMaketitlepage` command.

Since the `\author` command consists of only one field, we recommend to simply add linebreaks when declaring the authors if several people co-author a document. However, should you not use the IMT Atlantique style and would like to print the authors' name on one line in the default header, we introduced the `\imtaAuthorShort` command which sets the `\imtaTheAuthorShort` macro. By default, it is equal to the `\theauthor` command, but can be redefined to be on one line only.

The version of the document can be user-specified ausing the `\imtaVersion` command.

Moreover, you can also add one or several partner's logo on the front cover, next to the one of IMT Atlantique. In order to achieve this, you can simply use the `\imtaAddPartnerLogo` command in the preamble of your document. The logo will be resized so that its maximum dimension is equal to the corresponding dimension of the IMT Atlantique's logo, unless instructed otherwise.

The `imtaMakeCover` command outputs a cover as the last page of the document. This page will always be a left page in a two-side document.

In order to add beautiful quotes to your document, it is possible to use the `\imtaQuote` environment.

> *Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.*

This package depends upon a number of external packages. The use of these is explained hereafter, and the parameters each is used with are specified as well. Furthermore, a code snippet is presented, that shows the import line and the settings of the corresponding package.

The `geometry` package provides ways to act on the document's format. This package defines a A4 format, with two-centimeter margins, and a top margin of an extra centimeter for the header.

```
\RequirePackage[a4paper, margin=2cm, top=3cm]{geometry}
```

The `graphicx` package lets input graphics and pictures into the document.

```
\RequirePackage{graphicx}
```

The `fontenc` package declares an encoding for the output font. The `imta_core` package uses a latin font whose encoding is `T1`.

```
\RequirePackage[T1]{fontenc}
```

The `hyperref` package helps typeset hypertext links. The `hidelinks` option hides the links, but keeps them clickable. To output a hypertext link, use the `\hyperref` command.

```
\RequirePackage[hidelinks]{hyperref}
```

The `inputenc` package manages the input format. For uniformization purpose, the `imta` package is written for a use with Unicode. Therefore, it is used with the `utf8` option.

```
\RequirePackage[utf8]{inputenc}
```

The `fancyhdr` package is used for customizing the header and the footer. In the default style, the body pages have the `fancy` style, that is defined as follows:

```
\pagestyle{fancy}
\fancyhead{}
\fancyfoot{}
\fancyhead[L]{\thetitle}
\fancyhead[R]{\imtaTheAuthorShort}
\fancyfoot[C]{\thepage}
```

A blank style, `imtaFirstpage`, is defined to remove the header and the footer on the first page.

```
\fancypagestyle{imtaFirstpage}{
    \fancyhf{}
    \renewcommand{\headrulewidth}{0pt}
}
```

```
\RequirePackage{fancyhdr}
```

The `tikz` package is used to create the decorative figures on the front cover and on the new chapter pages. It is also used to recreate the IMT Atlantique logo, which can be used thanks to the `\imtaLogoTikz {<width>}` command.

```
\RequirePackage{tikz}
```

The `titlesec` package allows to easily redefine a section title style, using various commands such as `\titleformat` and `\titlespacing`.

```
\RequirePackage{titlesec}
```

The `titling` package creates automated front cover or title pages. In this package, we provide a simple wrapper with respect to the IMT Atlantique colour code and additionnal capabilities.

```
\RequirePackage{titling}
```

The `anyfontsize` package provides various tools to modify the formatting of sections and chapters, using the `\chapterfont` and `\sectionfont` commands for instance.

```
\RequirePackage{secsty}
```

The `enumitem` package allows to modify various properties of lists, notably its bullets. This was used with the `pifont` package to produce the IMT Atlantique list rendering.

```
\RequirePackage{enumitem}
```

Similarly to `fontawesome`, the `pifont` package offers multiple special characters and symbols. It is used along `enumitem` to customize the lists rendering by redefining the bullets. It is also used to display the large quote symbols in the `\imtaQuote` environment.

```
    \RequirePackage{pifont}
```

The `mdframed` package creates various frames and boxes with a lot of customization options. It is used for the `imtaQuote` environment.

```
    \RequirePackage{mdframed}
```

- `\imtaQuote` displays a quote in a beautiful frame as shown in section 6.2.
- `\subtitle {<subtitle>}` defines a subtitle for the document, which is displayed on the front cover when using `\imtaMaketitlepage`.
- `\imtaSuperviser {<name>}` defines a new field for the supervisers. It works in a similar fashion to the `\author` command and is displayed on the front cover when using `\imtaMaketitlepage`.
- `\imtaVersion {<version>}` defines the version of the document which is displayed on the front cover.
- `\imtaAddPartnerLogo [<options>]{<filepath>}` adds a partner's logo next to the one of IMT Atlantique on the front cover when using `\imtaMaketitlepage`. By default, its largest dimension is set to the corresponding dimension of the IMT Atlantique's logo, but it can be overrided thanks to the `<options>` field, which is used as options for `\includegraphics`. It supports the insertion of multiple logos, adding a linebreak whenever necessary.
- `\imtaMaketitlepage` displays the IMT Atlantique styled front cover on the current page.
- `\imtaAuthorShort {<shortened names>}` defines shortened author names for the header in case the IMT Atlantique style is not enabled throughout the whole document. It works in a similar fashion to the `\author` command.
- `\imtaLogo {<width>}` inserts the IMT Atlantique's logo with the desired dimension by importing the `imta_logo.pdf` document.
- `\imtaLogoTikz {<width}` inserts the TikZ version of the IMT Atlantique's logo with the desired dimension. This removes the dependency to the `imta_logo.pdf` file.

- `\imtaSetIMTStyle` enables the IMT Atlantique document styling throughout the whole document.
- `\subsubsubsection {<title>}` defines another level of sectioning, which can be useful for appendices.
- `\imtaQuestionCounter` creates a new subsection with an independant question counter (useful for a practical report).
- `\imtaQuestionReset` resets the question counter.
- `\imtaMakeCover` displays the IMT Atlantique styled back cover on the current page. If the document is in `twoside` mode, this command will insert a blank page if necessary, to ensure the last page is on the left side.

While the `imta_core` package provides a LaTeX template that satisfies the IMT Atlantique corporate identity rules[2] the `imta_extra` package provides the user some slightly more advanced or specific features.

Since it is a package, it can be used for a variety of classes and geometry. It has been primarily designed to be used in `article` and `report` documents, either with `oneside`/`twoside` or `onecolumn`/`twocolumn` geometries. If you are not sure how to start, we invite you to have a look at the `skeleton` file provided with this template.

In order to use this template, you need to have a TeX distribution installed, copy this file in your working directory and add the `\usepackage {imta_extra}` command in the preamble of your top document.

If you would prefer to properly install the package once it for all instead of having multiple copies, you can do so by manually installing it. Since this can be a little tedious for a recent LaTeX user, we introduced a custom Python script. Simply run the `imta_install.py` script and everything should be alright. This script supports Python 2 and 3, and both MikTeX and TeXLive environments.

Since `imta_extra` uses the `minted` package for code colouring, two extra steps are necessary to fully resolve dependencies. First, you need to have `Pygmentize` installed. If this is not the case, you can do it using Python utilitary `pip` by executing `pip install Pygments`. Finally, you also need to build your document with the correct options. If you are using pdfLaTeX as your compiler, you need to add the `--shell-escape` option.

Note that this template is compatible with at least TeXLive and MiKTeXdistributions, although we have noticed some issues might happen when trying to resolve dependencies with the latter. If so, then try to update your distribution using your MikTeXmanager. [3]

When writing a technical report, it can be useful to write some code with syntax highlighting, which moreover matches the template colour code. With this idea in mind, we introduced two custom environments, based on the `minted` package: `\imtaCode {<language>}` and `\imtaConsole`. This means they are both `\verbatim`-based environments (ie: the text is not interpreted by LaTeX in any way).

```
\begin{imtaCode}{python}
def mac(a, b, c = 0):
    # multiply-accumulate function
    return a*b + c
\end{imtaCode}
```

```
def mac(a, b, c = 0):
    # multiply-accumulate function
    return a*b + c
```

Figura 6.3: Example of the `imtaCode` environment (left) and its rendering (right)

```
\begin{imtaConsole}
    pwd
    grep -R "lorem" *
\end{imtaConsole}
```

```
pwd
grep -R "lorem" *
```

Figura 6.4: Example of the `imtaConsole` environment (left) and its rendering (right)

To further ease this process and allow the user to import code from external documents (just as figures), two commands have been introduced: `\imtaCodeFromFile {<file>}{<language>}` and `\imtaConsoleFromFile {<file>}`.

[2] See *Charte Graphique* on the IMT Atlantique's intranet for more details.
[3] See *General Guide to Installing Packages with MikTeX Package Manager* on TeXStack Exchange for more details.

In long documents, in can be useful to structure the list of figures or tables by printing the highest section (chapter or section) containing the items. In order to achieve this, the `figure` and `table` environments have been redefined, as well as the `chapter` and `section` commands. The default behavior is thus to print the upper level section in the list of figures or tables. If you would like to disable this functionalities, you can do it by using the `nouppersectioninlof` or `nouppersectioninlot` option when loading the package. Simply type `\usepackage [<options>]{imta_extra}` in the preamble.

In order to ease the page numbering and differentiate the preamble of your document (front cover, table of contents, list of figures, abstract, etc) from the corpus of your document, we introduced the `\frontmatter` and `\mainmatter`. They respectively set the page numbering style to roman or arabic.

The `mdframed` package creates various frames and boxes with a lot of customization options. It is used for the custom code listings.

```
1    \RequirePackage{mdframed}
```

The `minted` package provides powerful code listings tools. This package however requires the installation of the Python pakage `Pygments` as well as the compilation option `--shell-escape`. It is used for the custom code listings.

```
1    \RequirePackage{minted}
```

- `\imtaCode {<language>}` displays its content as syntax highlighted code according to the provided language and with line numbers on the side.
- `\imtaConsole` displays its content in a custom verbatim environment.
- `\imtaInlinecode {<language>}{<code>}` displays its content on the current line as syntax highlighted code according to the provided language.
- `\imtaCodeFromFile {<language>}{<path>}` displays the content of the given file as syntax highlighted code according to the provided language and with line numbers on the side.
- `\imtaConsoleFromFile {<path>}` displays the content of a given file in a custom verbatim environment.
- `\imtaFrontMatter` resets the page counter and changes its style to roman numerals.
- `\imtaMainMatter` resets the page counter and changes its style to arabic numerals.
- `nouppersectioninlof` disables the printing of the upper section in the list of figures
- `nouppersectioninlot` disables the printing of the upper section in the list of tables