

A project report on  
**SOME CONTRIBUTION TO L<sup>A</sup>T<sub>E</sub>X TO FACILITATE DOCUMENT  
FORMATTING IN A TECHNICAL INSTITUTE**

*submitted in partial fulfillment of the  
requirements for the degree of*

**Bachelor of Engineering**  
*in*  
**Information Technology**

*by*

<b>Shilpa S. Mahadik</b>	<b>T-08-38</b>
<b>Sadaf H. Desai</b>	<b>T-08-07</b>
<b>Sayali S. Kalyani</b>	<b>T-08-06</b>
<b>Bhagyashree V. Chavan</b>	<b>T-08-36</b>

*under the guidance of*  
**Mr. S. V. Jadhav**



Department of Information Technology  
Finolex Academy of Management and Technology, Ratnagiri  
April 2012

# CERTIFICATE

The report titled  
**SOME CONTRIBUTION TO L<sup>A</sup>T<sub>E</sub>X TO FACILITATE DOCUMENT  
FORMATTING IN A TECHNICAL INSTITUTE**

*duly completed by*

<b>Shilpa S. Mahadik</b>	<b>T-08-38</b>
<b>Sadaf H. Desai</b>	<b>T-08-07</b>
<b>Sayali S. Kalyani</b>	<b>T-08-06</b>
<b>Bhagyashree V. Chavan</b>	<b>T-08-36</b>

*as a partial fulfillment of the  
requirements for the degree of*

**Bachelor of Engineering**  
*in*  
**Information Technology**

*from*  
**The University of Mumbai**

Mr. S. V. Jadhav  
(Project Guide)

(Internal Examiner)

(External Examiner)

(H.O.D.)

(Principal)

Date:

Place: Department of Information Technology,  
Finolex Academy of Management and Technology, Ratnagiri

# Abstract

$\text{\LaTeX}$  is a document preparation system that enables the document writer to concentrate on the contents without bothering much about the formatting. The basic  $\text{\LaTeX}$  software system is available free of charge.

The aim of the project is to generate class or style files in  $\text{\LaTeX}$ , which can be used by students/faculty/staff of Finolex Academy of Management and Technology to format generally, used documents such as, project reports, notices, handouts, internal notes, etc. The classes in  $\text{\LaTeX}$  provide automation to improve consistency, productivity, and accuracy in creating and maintaining documents, thereby avoiding the inefficiencies of word processors.

# Contents

<b>Abstract</b>	<b>iii</b>
<b>Table of Contents</b>	<b>iv</b>
<b>List of Figures</b>	<b>vi</b>
<b>1 Project Overview</b>	<b>1</b>
<b>2 Introduction and Motivation</b>	<b>3</b>
2.1 An Introduction to L <sup>A</sup> T <sub>E</sub> X . . . . .	3
2.2 Why L <sup>A</sup> T <sub>E</sub> X? . . . . .	4
2.3 Required Parts of L <sup>A</sup> T <sub>E</sub> Xinput file . . . . .	4
2.4 Customizing L <sup>A</sup> T <sub>E</sub> X . . . . .	5
2.4.1 Creation of style file . . . . .	5
2.4.2 Creation of class file . . . . .	5
2.5 Motivation . . . . .	6
<b>3 Problem Statement</b>	<b>7</b>
<b>4 Implementation Details</b>	<b>8</b>
4.1 List of packages . . . . .	8
4.2 CollegeNotice.cls . . . . .	12
4.2.1 Using the class CollegeNotice . . . . .	12
4.2.2 What the CollegeNotice class does . . . . .	12
4.3 ProjectReport.cls . . . . .	13
4.3.1 Using the class ProjectReport . . . . .	13
4.3.2 What the ProjectReport class does . . . . .	13
<b>5 Technologies Used</b>	<b>16</b>
5.1 MikTeX - An Introduction . . . . .	16
5.2 Features of MikTeX . . . . .	16
5.3 Components of MikTeX . . . . .	18
5.4 Utilities in MikTeX . . . . .	18
<b>6 Project Timeline</b>	<b>20</b>
<b>7 Task Distribution</b>	<b>21</b>

<b>8 Conclusion and Future Work</b>	<b>22</b>
8.1 Conclusion . . . . .	22
8.2 Future Work . . . . .	22
<b>References</b>	<b>22</b>
<b>Acknowledgements</b>	<b>23</b>
<b>Appendix</b>	<b>24</b>

# List of Figures

6.1	Gantt Chart . . . . .	20
7.1	Work Breakdown Structure . . . . .	21

# Chapter 1

## Project Overview

$\text{\LaTeX}$  purpose is to simplify  $\text{\TeX}$  typesetting.  $\text{\LaTeX}$  has the provision of creating user-defined style and class files.

The project - Some contribution to  $\text{\LaTeX}$  to facilitate document formatting in a technical institute - attempts to create class files for project reports and notices as per technical specifications.

A standard project report includes key points - title page, certificate page, abstract, contents, references, appendix and acknowledgements. The title page in a standard project report has key fields - project title, degree or course, college name, department name, name and registration number of the students involved in the project, name of the project guide and logo of college. In addition to these fields, the certificate page in a standard project report includes the university to which the college is affiliated.

Care has been taken that these above mentioned fields in project report will be typeset as per the standards of the college Finolex Academy of Management and Technology. These fields are not limited to their default values, but a provision has been made for users outside this college to use these class files by changing default parameters for their project reports.

For this purpose, the project facilitates the typesetting by making default specifications for the page alignment, line spacing, font sizes, styles, headers, footers, figures, equations, etc. Thus the project helps in best possible way to reduce the user efforts in

---

documentation, by making it more handy and less time consuming.

Pre-requisites for project are - at a minimum, you'll need a  $\text{T}_{\text{E}}\text{X}$  distribution, a good text editor and a PDF viewer.



# Chapter 2

## Introduction and Motivation

The generally used documents such as project reports, notices, internal notes etc. in any technical institute are expected to be submitted in a standard specified format. The commonly used editing tools for this purpose are Microsoft Word, notepad, wordpad, etc. Taking into consideration various formatting constraints namely alignment, font styles, paragraphs, sections, subsections, etc. becomes a bit tedious using MS Word or other tools. Also maintaining the subscripts and superscripts to obtain the various mathematical equations becomes difficult and time consuming.

To overcome this drawback we have  $\text{\LaTeX}$ , a documentation preparation system that enables the document writer to concentrate on the contents of their text, without bothering too much about the formatting of it.

### 2.1 An Introduction to $\text{\LaTeX}$

$\text{\LaTeX}$  is a document preparation system for high-quality typesetting. It is most often used for medium-to-large technical or scientific documents but it can be used for almost any form of publishing.

$\text{\LaTeX}$  is not a word processor.  $\text{\LaTeX}$  is based on the idea that authors should be able to focus on the content of what they are writing without being distracted by its visual presentation. In preparing a  $\text{\LaTeX}$  document, the author specifies the logical structure

and lets the  $\LaTeX$  system worry about the presentation of these structures. It therefore encourages the separation of layout from content while still allowing manual typesetting adjustments where needed.

$\LaTeX$  is based on Donald E. Knuth's  $\TeX$  typesetting language or certain extensions.  $\LaTeX$  was first developed in 1985 by Leslie Lamport.

## 2.2 Why $\LaTeX$ ?

What you see is what you get (WYSIWYG) programs make it easy to put text wherever you want in whatever size and style of type you want, i.e., WYSIWYG programs offer visual design. The visual design is fine for short, simple documents like letters and memos. The visual design is not good for more complex documents such as scientific papers. For this purpose, we use  $\LaTeX$  that offers logical design.

$\LaTeX$  is intended to provide a high-level language that accesses the power of  $\TeX$ .  $\LaTeX$  comprises a collection of  $\TeX$  macros and a program to process  $\LaTeX$  documents. Because the  $\TeX$  formatting commands are very low-level, it is usually much simpler for end-users to use  $\LaTeX$ . As  $\LaTeX$  is distributed under the terms of the  $\LaTeX$  Project Public License (LPPL),  $\LaTeX$  is free software.

## 2.3 Required Parts of $\LaTeX$ Input file

A few commands must appear in every  $\LaTeX$  input file in a certain order. They are:

```
\documentclass{classname}
\begin{document}
\end{document}
```

The `documentstyle` has a required argument `stylename` to select an overall type-setting style for the document; the one normally used is `article` (there are also `book`, `report`, `letter`, and `memo`). It also has an optional argument to select 11pt or 12pt normal type

size (10pt is the default size). The actual text of user document and associated commands go between the begin and end commands.

## 2.4 Customizing L<sup>A</sup>T<sub>E</sub>X

There are situations where L<sup>A</sup>T<sub>E</sub>X does not provide a command or environment that matches user needs, or the output produced by some existing command may not meet user requirements. To add your own commands, use

```
\newcommand{name}[num]{definition}
```

The command requires two arguments: the name of the command you want to create, and the definition of the command. The num argument in square brackets is optional.

### 2.4.1 Creation of style file

Typically, a style sheet is specified at the beginning of document. This style sheet applies to the entire document. To create your own style file, at the very beginning of the text document just write,

```
\ProvidesPackage{mypack}
```

where mypack is the name of package. Write whatever you want in it using all the L<sup>A</sup>T<sub>E</sub>X commands you know. The style file should have the same name as that of the package name. Save this style file with extension .sty. Now, it is necessary to import this style file in your T<sub>E</sub>X document which can be done using following command.

```
\usepackage{mypack}
```

where mypack is the style file you have created earlier.

### 2.4.2 Creation of class file

Create a simple file called myclass.cls and open it with any text editor, at the very beginning of the document just write

```
\NeedsTeXFormat{LaTeX2e}
```

```
\ProvidesClass{<class-name>}[<date><other information>]
```

Write whatever you want in it using all the  $\LaTeX$  commands you know. Import your new class with the known command

```
\documentclass{myclass}
```

The file `myclass.sty` and the  $\LaTeX$  source you are compiling must be in the same directory.

## 2.5 Motivation

The existing word processors have several limitations which can be overcome by  $\LaTeX$ .

The main advantages of  $\LaTeX$  over normal word processors are the following:

- Professionally crafted layouts are available, which make a document really look as if “printed”.
- The typesetting of mathematical formulae is supported in a convenient way.
- Users only need to learn a few easy-to-understand commands that specify the logical structure of a document. They almost never need to tinker with the actual layout of the document.
- Even complex structures such as footnotes, references, table of contents, and bibliographies can be generated easily.
- Free add-on packages exist for many typographical tasks not directly supported by basic  $\LaTeX$ . For example, packages are available to include PostScript graphics or to typeset bibliographies conforming to exact standards.
- $\LaTeX$  encourages authors to write well-structured texts by specifying structure.

# Chapter 3

## Problem Statement

The proposed system attempts to use  $\text{\LaTeX}$  to make documentation of generally used documents like project thesis, reports, notices, etc. in the technical institutes a bit easier.

The proposed system will provide class or style files for a technical institute to write

1. notices
2. project reports

The user simply needs to include the package name in his file and use the commands defined in the package to format his documents. Thus, the proposed system will reduce the efforts of the user in formatting the documents.

# Chapter 4

## Implementation Details

### 4.1 List of packages

The list of packages required to implement this project are:

1. `graphicx`

This package is an extension of the standard graphics bundle and provides a way to include repeated postscript graphics (`ps`, `eps`) only once in a postscript document. This provides a way to get smaller postscript documents when having, for instance, a logo on every page. This package only works when post-processed with `dvips`.

2. `epstopdf`

The package adds support for EPS files in the `graphicx` package when running under PDF $\TeX$ . If an EPS graphic is detected, the package spawns a process to convert the EPS to PDF, using the script `epstopdf`. This of course requires that shell escape is enabled for the PDF $\TeX$  run.

3. `ifthen`

Conditional commands in  $\LaTeX$  documents

The package's basic command is

```
\ifthenelse
```

which can use a wide array of tests. Package also provides a simple loop command

`\whiledo`

4. xifthen

Extended conditional commands

This package extends the ifthen package by implementing new commands to go within the first argument of

`\ifthenelse`

to test whether a string is void or not, if a command is defined or equivalent to another. The package also offers the possibility to make use of complex expressions, together with the ability of defining new commands to handle complex tests.

5. changepage

Margin adjustment and detection of odd/even pages

The package provides commands to change the page layout in the middle of a document, and to robustly check for typesetting on odd or even pages. Instructions for use are at the end of the file. The package is an extraction of code from the memoir class, whose user interface it shares. It is intended this package will eventually replace the chngpage package, which is distributed with the package.

6. geometry

Flexible and complete interface to document dimensions

The package provides an easy and flexible user interface to customize page layout, implementing auto-centering and auto-balancing mechanisms so that the users have only to give the least description for the page layout.

7. fontenc

Standard package for selecting font encodings

The package allows the user to select font encodings, and for each encoding provides an interface to font-encoding-specific commands for each font. Its most powerful effect is to enable hyphenation to operate on texts containing any character in the font.

#### 8. mathptmx

This package defines Adobe Times Roman (or equivalent) as default text font, and provides maths support using glyphs from the Symbol, Chancery and Computer Modern fonts together with letters, etc., from Times Roman. It supersedes both the original times and the mathptm packages.

#### 9. fancyhdr

Creating headers and footers in  $\text{\LaTeX}$  can be done by using the package “fancyhdr”.

First of all, you need to tell  $\text{\LaTeX}$  to use the package:

```
\usepackage{fancyhdr}
```

and change the style from plain to fancy:

```
\pagestyle{fancy}
```

You will now get the default fancy pagestyle which adds a line at the top of every page, except for some exceptions (title-page, abstract, new chapter in report). Even though fancyhdr has a default page style, you are free to define headers/footers yourself.

#### 10. caption

Customising captions in floating environments

The caption package provides many ways to customise the captions in floating environments like figure and table, and cooperates with many other packages. Facilities include rotating captions, sideways captions, and continued captions (for tables or figures that come in several parts).



## 11. appendix

Extra control of appendices

The appendix package provides various ways of formatting the titles of appendices. Also (sub)appendices environments are provided that can be used, for example, for per chapter/section appendices.

## 12. beamer

A  $\LaTeX$  class for producing presentations and slides

The beamer class can be used for producing slides. It does not need any external programs and can directly produce a presentation using pdf $\LaTeX$ . Beamer uses pgf for pdf/ps independent graphics. Frames are created using

```
\frame{...}
```

Beamer supports bibliographies, appendices and transitions. Short versions of title, authors, institute can also be specified as optional parameters.

```
\plainframe{}
```

allows a picture, for example, to fill the whole frame. Support figure and table environments, transparency effects, transduration command, animation commands, pauses environment.

## 13. hyperref

Extensive support for hypertext in  $\LaTeX$

The hyperref package is used to handle cross-referencing commands in  $\LaTeX$  to produce hypertext links in the document. The package is distributed with the backref and nameref packages, which make use of the facilities of hyperref. The package depends on the author's kvoptions, ltxcmds and refcount packages.

## 4.2 CollegeNotice.cls

### 4.2.1 Using the class CollegeNotice

To use the CollegeNotice class, make sure that the CollegeNotice.cls file is on your TEX-INPUTS search path and use the following command at the start of your input file:

```
\documentclass{ CollegeNotice}
```

### 4.2.2 What the CollegeNotice class does

The CollegeNotice class provides users of Finolex Academy of Management and Technology(FAMT) to prepare a notice according to standard norms.

The key features of the class are - By default this class uses FAMT as default college and Information Technology as default department, which can be changed as per user requirements.

**Default values for the commands are:**

College - Finolex Academy of Management and Technology

Department - Information Technology

Issuedate -  $\LaTeX$  takes current date from the system

Desigantion - Head of Department

**Mandatory values user needs to provide are:**

Subject of notice

Content of notice

**To overwrite default values commands to be used are:**

`\College` Specify the name of the college, if it is other than FAMT

`\Department` Specify the name of the department, if it is other than IT

`\Issuedate` The date when the notice is to be issued

`\Name` The name of the person issuing the notice

`\Designation` The name of the person issuing the notice

## 4.3 ProjectReport.cls

### 4.3.1 Using the class ProjectReport

To use the ProjectReport class, make sure that the ProjectReport.cls file is on your TEX-INPUTS search path and use the following command at the start of your input file:

```
\documentclass{ProjectReport}
```

### 4.3.2 What the ProjectReport class does

The ProjectReport class is a modified version of the standard L<sup>A</sup>T<sub>E</sub>XREPORT class that is accepted for use with Finolex Academy of Management and Technology (FAMT) dissertations theses. The ProjectReport class allows users to concentrate on the content of report instead of formatting.

**The key features of the class are:**

1. The title page and certificate page is designed to fulfill the specifications of FAMT institute.
2. Margins are 1.3 inches from the left, 1.1 inches from right, 1.4 inches from top and 1.1 inches from bottom.
3. Uses 12 point by default.
4. Page numbers are in the center at the bottom for all pages.
5. Complete, correct front matter for ProjectReport can be generated.

**Default values for commands are:**

College -Finolex Academy of Management and Technology

Department - Information Technology

University- Mumbai University

Month - April

Degree - Bachelor of Engineering

**Mandatory values user needs to provide are:**

Name of students

Registration number of students

Name of Guide

Title of Project Report

**To overwrite default values, commands to be used are:**

`\College`

`\Department`

`\University`

`\Degree`

`\monthnum`

`\Year`

**Layout constraints:**

1. Margins - Using geometry package, margins are set to 1.3 inches from the left, 1.1 inches from right, 1.4 inches from top and 1.1 inches from bottom.
2. Headers and Footers - Using fancyhdr package, page numbers are set in the center at the bottom for all pages and section name in header at the right corner.
3. Fonts - Using fontenc, mathptmx package, the default font size is set to 12, font family is set to Times New Roman.

**Front matter:**

The title page, certificate page, abstract, table of contents, lists of figures MUST be put inside the "frontmatter" environment, which ensures that page-numbering is handled properly.

**College Logo and other Figures:**

The College Logo is provided in eps format so that it cannot be edited. To include Logo.eps into the Project Report the package eps2pdf is used.

The figures related to project should be included in figure environment to ensure that numbering is handled properly.

# Chapter 5

## Technologies Used

The recommended  $\text{T}_{\text{E}}\text{X}$  distributions are:

- $\text{T}_{\text{E}}\text{X}$ Live is a major  $\text{T}_{\text{E}}\text{X}$  distribution for Unix/Linux, Mac OS and Windows.
- MiKTeX is a Windows-specific distribution.
- MacTeX is a Mac OS-specific distribution based on  $\text{T}_{\text{E}}\text{X}$ Live.

In project, MikTeX 2.9 has been used.

### 5.1 MikTeX - An Introduction

MiKTeX (pronounced mick-tech) is a typesetting system for Microsoft Windows that is developed by Christian Schenk. MiKTeX provides the tools necessary to prepare documents using the  $\text{T}_{\text{E}}\text{X}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  markup language. The current version of MiKTeX is 2.9. Since version 2.7, MiKTeX has compatibility with Windows 7. MiKTeX is currently being ported to GNU/Linux, with a beta version being available for testing.

### 5.2 Features of MikTeX

1. Easy to install

It is very easy to install MiKTeX. The MiKTeX Setup Wizard guides you through

the installation process. You don't have to be a computer expert.

2. Complete

A complete set of additions (programs, styles, fonts, ...) are available to help you typeset your documents.

3. Integrated package management

MiKTeX's integrated package manager installs missing components from the Internet, if required.

For example, the  $\LaTeX$  directive

```
\usepackage{memoir}
```

would trigger the installer, if the memoir package is not already installed.

This unique feature allows you to keep your computer clutter-free: only install packages you actually need.

4. Network friendly

MiKTeX can be run directly from a shared and read-only network directory

5. Enhanced  $\TeX$  compiler capabilities

6. Always up-to-date

The MiKTeX Update Wizard helps you to keep your MiKTeX system up-to-date by installing the latest package updates.

7. Fast previewer

The  $\TeX$ output viewer Yap allows for an optimized edit-compile-view cycle (if the  $\TeX$ authoring system in use supports source specials).

8. Well maintained

The MiKTeX package repository is updated regularly

9. Open source

MiKTeX is open source. You are welcome to redistribute MiKTeX under certain conditions.

## 5.3 Components of MikTeX

The MiKTeX distribution consists of the following components:

- $\text{\TeX}$ , METAFONT, TeXware, METAFONTware, Computer Modern Fonts-the base  $\text{\TeX}$ system
- pdfTeX, XeTeX, Omega, NTS-various  $\text{\TeX}$ derivatives
- MetaPost-a METAFONT derivative for the creation of PostScript figures
- Dvipdfm-converts  $\text{\TeX}$ output into PDF documents
- macro packages-almost all free  $\text{\TeX}$ macro packages
- fonts-almost all free fonts
- TeXify-a  $\text{\TeX}$ compiler driver
- MiKTeX Options-assists in configuring MiKTeX
- MiKTeX Update Wizard-assists in keeping the MiKTeX system up-to-date
- lots of utilities tools for the creation of bibliographies & indexes, PostScript utilities

## 5.4 Utilities in MikTeX

A few utilities required in MiKTeX packaging are:

1. cabextract

cabextract is used internally (by MPC) to extract files from cabinet files. It is a part of the Cygwin distribution for Windows.



2. lzma

lzma is used internally (by MPC) to extract files from .tar.lzma files. It is a part of the Cygwin distribution for Windows.

3. md5walk

MD5Walk is used to calculate the fingerprint of a package. The source code of MD5Walk can be obtained from the MiKTeX project page.

4. mpc

The MiKTeX Package Creator (MPC) is used to maintain the contents of a custom package repository. The source code of MPC can be obtained from the MiKTeX project page.

5. rsync

rsync is used to initialize the custom package repository. It is a part of the Cygwin distribution for Windows.

6. tar

tar is used internally (by MPC) to create package archive files. It is a part of the Cygwin distribution for Windows.

All but MPC and MD5Walk are standard utilities which are available on all platforms (Windows, GNU/Linux, ...). The portable source code of mpc and md5walk can be obtained from the MiKTeX project page.

# Chapter 6

## Project Timeline

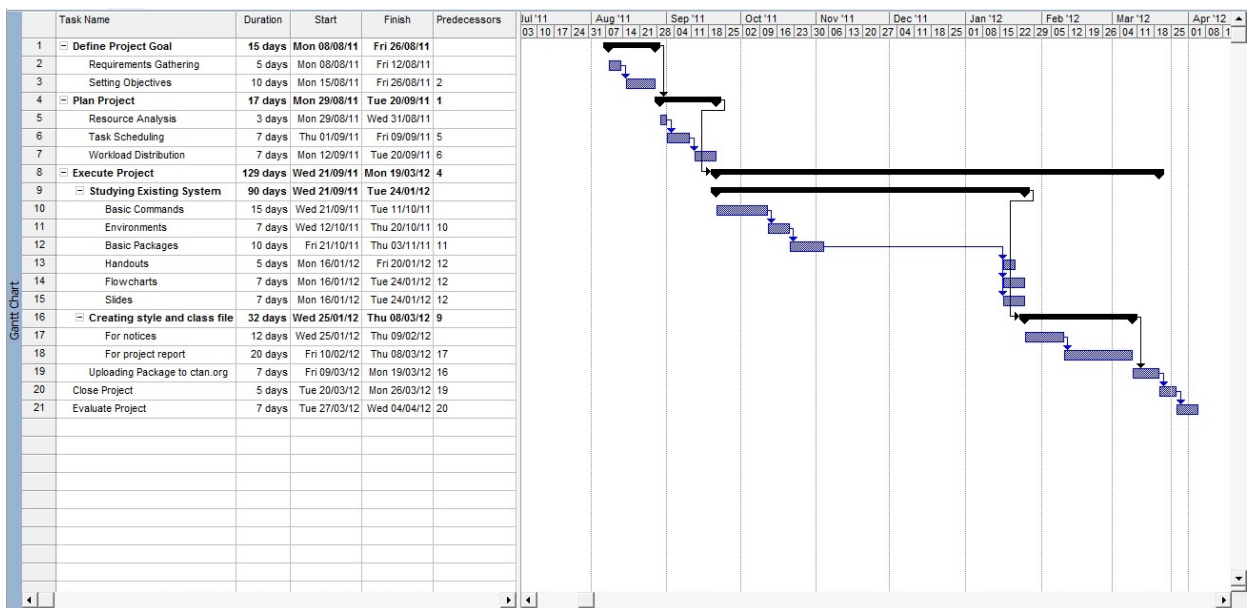
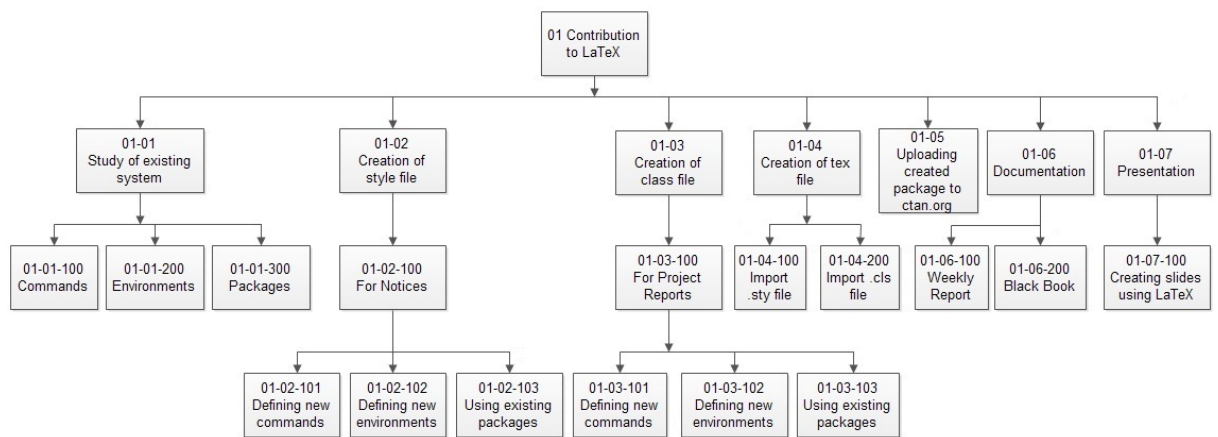


Figure 6.1 Gantt Chart

# Chapter 7

## Task Distribution



**Figure 7.1** Work Breakdown Structure

# Chapter 8

## Conclusion and Future Work

### 8.1 Conclusion

Thus our project “Some contribution to L<sup>A</sup>T<sub>E</sub>X to facilitate document formatting in technical institute” concentrates on the documentation of notices, project reports which includes project title page, certificate etc. The standard formatting constraints for these documents are defined in packages developed under this project, which will thus help the user to complete his/her work in stipulated time and making it less tedious.

### 8.2 Future Work

In order to make use of class files developed in the project, the basic requirement is that the user must have the sound knowledge of L<sup>A</sup>T<sub>E</sub>X. So to make it more handy, user interface can be developed which will take only the required values as input viz; the content of notice and the packages in the back-end will take care of formatting without the user having to know L<sup>A</sup>T<sub>E</sub>X commands.

# References

1. A document preparation system  $\text{\LaTeX}$ -Leslie Lamport
2. <http://tug.org/tutorials/tugindia/>
3. <http://www.andy-roberts.net/writing/latex>
4. <http://miktex.org>
5. <http://www.ctan.org/>

# Acknowledgements

The efforts taken in this project would not have been possible without the kind support and help of many individuals. We would like to extend sincere thanks to all of them.

We are highly indebted to Mr. S.V. Jadhav for his invaluable guidance and constant supervision as well as for providing necessary support in completing the project.

We would like to express gratitude towards Mr. Parag Abhyankar for his encouragement which helped us in completion of this project.

We would like to express our special gratitude and thanks to all those user groups and communities who have worked on open source document preparation system  $\text{\LaTeX}$  and have made available their work for our use.

Shilpa Mahadik

Sadaf Desai

Sayali Kalyani

Bhagyashree Chavan

# Appendix

It is .tex file of project report.

```
\documentclass{ProjectReport}
\Protitle{Title of Project}
\MemberA{Shilpa S. Mahadik} \RegA{T-08-38}
\MemberB{Sadaf H. Desai} \RegB{T-08-07}
\MemberC{Sayali S. Kalyani} \RegC{T-08-06}
\MemberD{Bhagyashree V. Chavan} \RegD{T-08-36}
\Guidename{Mr. S. V. Jadhav}
\begin{document}
\frontmatter
\makepreliminarypages
\chapter*{Abstract}
\tableofcontents
\listoffigures
\mainmatter
\chapter{Project Overview}
\chapter{Introduction and Motivation}
\chapter{Problem Statement}
\chapter{Implementation Details}
\chapter{Technologies Used}
\chapter{Project Timeline}
\chapter{Task Distribution}
```

```
\chapter{Conclusion and Future Work}  
\chapter*{References}  
\chapter*{Acknowledgements}  
\end{document}
```