

```

chapter = format = , nameformat = , titleformat =
, number = 0, beforeskip = 24bp, afterskip =
18bp, , section = format = , beforeskip = 24bp, afterskip =
6bp, , subsection = format = , beforeskip = 12bp, afterskip = 6bp,
, subsubsection = format = , beforeskip = 12bp, afterskip = 6bp, sec-
tion/format += subsection/format +=

```

软件工程作业管理系统 概要设计

	人员	日期
拟制	张三 李四 王五	yyyy-mm-dd
评审人	•	yyyy-mm-dd
批准	•	yyyy-mm-dd
签发	•	yyyy-mm-dd

chapter/format +=

摘 要

本文是软件工程需求规格说明书模板，修改自于中国科学技术大学本硕博毕业论文 L^AT_EX 模板示例文件，该模板由 zepinglee 和 seisman 创建，遵循中国科学技术大学的论文写作规范，适用于撰写学士、硕士和博士学位论文。

本文档最后一章演示如何使用 L^AT_EX 的一些基本命令以及本模板提供的一些特殊功能，模板的选项及详细用法请参考模板说明文档 `ustcthesis.pdf`。请在提交之前把最后一掌实例注释掉。

关键词： 软件工程 中国科学技术大学 学位论文 L^AT_EX 通用模板 学士 硕士 博士 示例文档 模板说明文档

表 1 缩略词清单

缩略语	英文全名	中文解释
c	d	e

目 录

摘要	
第 1 章 引言	6
1.1 编写目的	6
1.2 项目背景	6
1.3 术语	6
第 2 章 任务概述	8
2.1 目标	8
2.1.1 总体目标	8
2.1.2 高级目标	8
2.2 开发与运行环境	8
2.2.1 开发环境的配置	8
2.2.2 测试环境的配置	8
2.2.3 运行环境的配置	8
2.3 需求概述	8
2.4 条件与限制	10
2.4.1 硬件约束	10
2.4.2 技术限制	10
第 3 章 总体设计	11
3.1 软件描述	11
3.2 处理流程	11
3.2.1 总体流程	11
3.2.2 系统基本流程	11
3.2.3 客户端基本流程	11
3.2.4 服务器端基本流程	11
3.2.5 功能 1 具体流程	11
3.2.6 功能 2 具体流程	11

3.2.7 功能 3 具体流程	12
3.3 功能结构设计	12
3.3.1 整体结构	12
3.3.2 用户端结构	12
3.3.3 服务器端结构	12
3.3.4 后台数据库维护模块结构	12
3.4 功能需求与程序代码的关系	12
第 4 章 接口设计	14
4.1 外部接口	14
4.1.1 支付宝接口	14
4.2 内部接口	14
第 5 章 数据结构设计	15
5.1 逻辑结构设计	15
5.1.1 用户管理系统数据结构设计	15
5.1.2 客户端数据结构	15
5.1.3 用户端数据结构	15
5.2 物理结构设计	15
5.3 数据结构与程序模块的关系	15
第 6 章 数据库设计	16
6.1 数据库环境说明	16
6.2 数据库的命名规则	16
6.3 逻辑设计	16
6.4 物理设计	16
6.4.1 数据库产品	16
6.4.2 实体属性、类型、精度	17
6.5 安全性设计	17
6.6 数据库管理与维护说明	17
第 7 章 界面设计	18
7.1 客户端界面	18
7.2 服务器端界面	18
7.3 登录界面	18
7.4 xxx 功能界面	18

第 8 章 出错处理设计	19
8.1 数据库出错处理	19
8.2 某模块失效处理	19
第 9 章 安全保密设计	20
第 10 章 维护设计	21
第 11 章 图片	22
11.1 示例	22
11.2 带图注的图	22
第 12 章 表格	24
12.1 A Simple Table	24
12.2 长表格	24
第 13 章 算法环境	26
第 14 章 代码环境	28
第 15 章 引用文献标注	29
15.1 著者-出版年制标注法	29
15.2 顺序编码制标注法	29
15.3 其他形式的标注	30
参考文献	31

图目录

11.1 测试图片	22
11.2 带图注的图片	23

表目录

1 缩略词清单	
1.1 术语表	7
2.1 开发环境的配置	9
2.2 测试环境的配置	9
2.3 运行环境的配置	10
3.1 功能需求与程序代码的关系表	13
5.1 数据结构与程序代码的关系表	15
6.1 用户数据表 Users 设计	17
6.2 联系人表 Links 设计	17
12.1 这里是表的标题	24
12.2 长表格演示	24

第一章 引言

1.1 编写目的

在本项目的前一阶段，也就是需求分析阶段，已经将系统用户对本系统的需求做了详细的阐述，这些用户需求已经在上一阶段中对不同用户所提出的不同功能，实现的各种效果做了调研工作，并在需求规格说明书中得到详尽得叙述及阐明。

本阶段已在系统的需求分析的基础上，对即时聊天工具做概要设计。主要解决了实现该系统需求的程序模块设计问题。包括如何把该系统划分成若干个模块、决定各个模块之间的接口、模块之间传递的信息，以及数据结构、模块结构的设计等。在以下的概要设计报告中将对在本阶段中对系统所做的所有概要设计进行详细的说明，在设计过程中起到了提纲挈领的作用。

在下一阶段的详细设计中，程序设计员可参考此概要设计报告，在概要设计即时聊天工具所做的模块结构设计的基础上，对系统进行详细设计。在以后的软件测试以及软件维护阶段也可参考此说明书，以便于了解在概要设计过程中所完成的各模块设计结构，或在修改时找出在本阶段设计的不足或错误。

本文档主要由开发人员撰写，意在指明即时通讯软件系统（以下简称通讯系统）的具体开发明细，记录下通信软件。本文档还可作为软件开发过程中的备案，为后续需求的提出给出参考。此外，非软件开发者可以将这个文件视为我们的即时通讯系统的开发纲要

1.2 项目背景

随着网络通信和信息技术的发展，现代社会人与人之间的交流越来越重要，同时也越来越便捷，并且出现了如 Facebook、QQ 等著名社交软件。于是我们组决定立足于当下标准，设计一款同样有着及时，便捷，潮流的通讯软件。

1.3 术语

[列出本文档中所用到的专门术语的定义和外文缩写的原词组]

表 1.1 术语表

缩写、术语	解释
c	d

第二章 任务概述

本系统的目标是实现一个即时通信系统，包括客户端、服务器端两个部分。

客户端面向 Android 或 windows 用户，为用户提供即时的文字、语言聊天，群聊等基本功能，考虑实现各种高级功能。

2.1 目标

2.1.1 总体目标

实现实时通信系统的基本功能，实现需求规格说明书中所描述的功能需求、性能需求，以及实现实时聊天功能和群聊功能，并且保证系统的健壮性和数据安全。

2.1.2 高级目标

提供稳定的视频聊天功能。

同时还能支持动态分享，皮肤自定义等个性化功能。

2.2 开发与运行环境

2.2.1 开发环境的配置

2.2.2 测试环境的配置

2.2.3 运行环境的配置

2.3 需求概述

功能需求包括：通讯模式 – 用户一对一通讯 – 群聊 – 随机匹配聊天 • 通讯方式 – 文字即时通讯 – 图片、视频、音频、链接、文件等多种媒体的发送 – 语音、视频及时通讯 – 加密文字及时通讯 • 用户控制 – 注册/注销账号 – 添加/删除联系人 – 加入/退出/创建群聊

表 2.1 开发环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	windows .net 版本 >4.0	windows .net 版本 >4.0
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

表 2.2 测试环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version ≥ 4.10) GNU gcc (version $\geq 6.3.1$)	Linux (kernel version ≥ 3.10) GNU gcc (version ≥ 5.4)
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

表 2.3 运行环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version ≥ 4.10) GNU gcc (version $\geq 6.3.1$)	Linux (kernel version ≥ 3.10) GNU gcc (version ≥ 5.4)
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

2.4 条件与限制

2.4.1 硬件约束

1. Android 平台：软件本身大小 $< 100\text{MB}$ ；本地聊天记录文件不大于 1G ，并且有着自动清除过时的聊天记录功能；文本通信延迟 $< 100\text{ms}$ ；内存占用 $< 100\text{MB}$ ；流量消耗不超过通信本身数据量的 1.1 倍；电量消耗不超过机体待机时的两倍

2. Windoes 平台：约束较少，主要考虑的磁盘和内存约束与 Android 一致

2.4.2 技术限制

1. 考虑到服务器网络带宽问题，视频聊天的清晰度最高维持在 360P ；

2. 由于技术上的限制，除去文本类的通信记录都不会被保存；

3. 用户无法在不同的客户端上共享聊天记录；

4. 用户无法在周日凌晨 2:00 至 4:00 使用通讯软件的聊天功能因为这个时候应该是维护服务器的时候；

5. 由于服务器磁盘大小的限制，用户传输文件的大小有着 $< 8\text{MB}$ 的限制

第三章 总体设计

3.1 软件描述

系统包括前台和后台两个部分。

前台主要功能是：给用户提供交互的接口，用于各种功能。后台主要功能是：处理用户发出的请求，后台记录用户产生的数据等等。

3.2 处理流程

3.2.1 总体流程

此处应当有一个图和对应的描述。

3.2.2 系统基本流程

此处应当有一个图和对应的描述。

3.2.3 客户端基本流程

用户登录 -> 显示消息列表 -> 选择具体功能 -> 聊天或者等待或者退出

3.2.4 服务器端基本流程

3.2.5 功能 1 具体流程

1. 用户注册或登录

在打开应用的第一个页面，会有用户名和密码的输入框，在正确输入用户名和密码之后点击 **sign in** 便可以登入。或者可以点击另外一个 **sign up** 的按钮，创建一个用户名没有重复的账户之后输入验证码便会通过，服务器数据更新之后便可以使用这个新的账户登录。

3.2.6 功能 2 具体流程

2. 单人聊天

已登录用户会进入一个新的功能页面（称为主页面）。在主页面首先在消息列表中选择已有对话，或者在联系人列表中选择联系人新建对话。

3.2.7 功能 3 具体流程

3. 大厅聊天

已登录用户在主页面选择大厅聊天，输入任何自己想说的并发送即可。要注意的是，在大厅说的所有话都会被所有在线用户看到。

3.3 功能结构设计

3.3.1 整体结构

此处应当有一个图和对应的描述。系统如果像微内核那样，划分成核心模块和若干个子系统，此处应当有图示及说明，然后后续几个节应当描述这几个子系统。如果系统像宏内核，那应当说明有哪些紧密联系的模块，并在后续几个节内描述这些模块。

3.3.2 用户端结构

此处应当有一个图和对应的描述。这只是举个例子。可能的内容包括用户端的具体模块、耦合情况等。

3.3.3 服务器端结构

此处应当有一个图和对应的描述。这只是举个例子。

3.3.4 后台数据库维护模块结构

此处应当有一个图和对应的描述。这只是举个例子。

3.4 功能需求与程序代码的关系

[此处指的是不同的需求分配到哪些模块去实现。可按不同的端拆分此表]

表 3.1 功能需求与程序代码的关系表

•	模块 1	模块 2	模块 3
需求 1	•	Y	•
需求 2	•	Y	•
需求 3	•	Y	•
需求 4	Y	•	•
需求 5	•	•	Y

注：各项功能需求的实现与各个程序模块的分配关系

第四章 接口设计

4.1 外部接口

比如说需要用到支付宝等外部支付系统，接口应当如何封装。

4.1.1 支付宝接口

详细讲述不同的接口（查询状态、支付交易、获取回执等）

4.2 内部接口

内部模块/系统之间的交互的接口。

第五章 数据结构设计

5.1 逻辑结构设计

5.1.1 用户管理系统数据结构设计

讲述本系统内需要什么数据结构。这指的是程序运行过程中维护的数据结构。只是举个例子，此处应和 3.3 一致。

5.1.2 客户端数据结构

5.1.3 用户端数据结构

5.2 物理结构设计

各数据结构无特殊物理结构要求。（如果有，比如说 hadoop 等，应当具体说明）

5.3 数据结构与程序模块的关系

[此处指的是不同的数据结构分配到哪些模块去实现。可按不同的端拆分此表]

表 5.1 数据结构与程序代码的关系表

•	模块 1	模块 2	模块 3
结构 1	•	Y	•
结构 2	•	Y	•
结构 3	•	Y	•
结构 4	Y	•	•
结构 5	•	•	Y

注：各项数据结构的实现与各个程序模块的分配关系

第六章 数据库设计

6.1 数据库环境说明

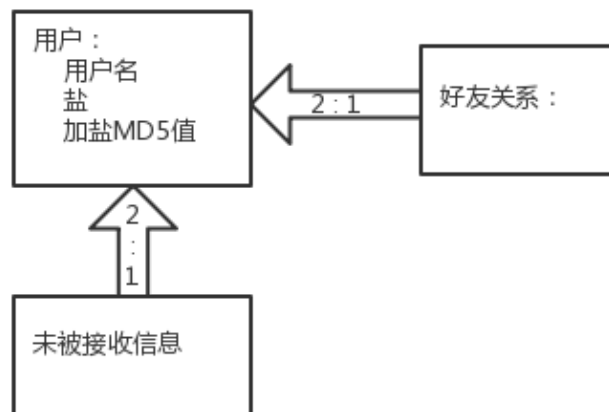
本聊天软件的服务器在云服务器上运行，操作系统是 Ubuntu16.04，所以本系统的数据系统采用 MySQL 数据库系统。

6.2 数据库的命名规则

不允许单词缩写，表名是复数形式。

6.3 逻辑设计

本数据库的 users 和 links 表能满足 BCNF 范式，但是 unreceived_message 表并不设主键，仅仅满足第一范式。



6.4 物理设计

6.4.1 数据库产品

使用 MySQL 数据库。根据目前本软件的使用情况，一台云服务器已经可以很好地满足所有需求，所以没有采用分布式数据库。

6.4.2 实体属性、类型、精度

6.4.2.1 客户数据表设计

表 6.1 用户数据表 Users 设计

字段名	类型	大小	说明	备注
uname	varchar	48	用户名	主键
md5	char	10	随机产生的加盐值	•
pw	char	32	用户的登录密码加盐的 MD5 值	•

注：用户数据表 Users 设计

6.4.2.2 联系人数据表设计

表 6.2 联系人表 Links 设计

字段名	类型	大小	说明	备注
uname_1	varchar	48	用户 1	主键，外键，引用 Users
uname_2	varchar	48	用户 2	主键，外键，引用 Users

注：联系人表 Links 设计

6.5 安全性设计

备份和容灾设计。

6.6 数据库管理与维护说明

对于数据库的维护，随时对数据库中的信息加以调试和保存备份。同样需要个工作人员进行系统的分析和用户的反馈，对系统进行升级以及功能的完善。同时保证系统安全有序的运行。

第七章 界面设计

7.1 客户端界面

此处应当有一个简略的图，重点是展示你与用户交互的逻辑。（processon 上画一个不花时间）

7.2 服务器端界面

此处应当有一个简略的图。

7.3 登录界面

此处应当有一个简略的图。

7.4 xxx 功能界面

此处应当有一个简略的图。

第八章 出错处理设计

8.1 数据库出错处理

多重备份时，应采取何种策略，先利用哪一份备份；系统是否暂停服务等。

8.2 某模块失效处理

是否整个系统暂停服务，还是维持最小服务状态、如何尽快恢复服务还是删库跑路等。

第九章 安全保密设计

可能的内容包括保密性、是否采取加密传输、密钥如何分发和管理等。

第十章 维护设计

可能的内容包括数据库的日常备份、压缩、维护等。

第十一章 图片

本章展示图片相关用法。

11.1 示例



图 11.1 测试图片

11.2 带图注的图



图 11.2 带图注的图片

注: the solid lines represent the time histogram of the spontaneous activities of an old monkey cell(gray) and a young monkey cell (black). The bin-width is 1

第十二章 表格

12.1 A Simple Table

表 12.1 这里是表的标题

a	b
c	d

注：这里是表的注释

12.2 长表格

表 12.2 长表格演示

名称	说明	备注
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC
AAAAAAAAAAAA	BBBBBBBBBBBB	CCCCCCCCCCCC

续下页

表 12.2 长表格演示 (续)

[illegible]

第十三章 算法环境

模板中使用 `algorithm2e` 宏包实现算法环境。关于该宏包的具体用法，请阅读宏包的官方文档。

```

Data: this text

Result: how to write algorithm with LATEX2ε

1 initialization;
2 while not at end of this document do
3   read current;
4   if understand then
5     go to next section;
6     current section becomes this one;
7   else
8     go back to the beginning of current section;
9   end
10 end

```

算法 13.1: 算法示例 1

```

input : A bitmap  $Im$  of size  $w \times l$ 
output: A partition of the bitmap

1 special treatment of the first line;
2 for  $i \leftarrow 2$  to  $l$  do
3   special treatment of the first element of line  $i$ ;
4   for  $j \leftarrow 2$  to  $w$  do
5      $\text{left} \leftarrow \text{FindCompress}(Im[i, j - 1]);$ 
6      $\text{up} \leftarrow \text{FindCompress}(Im[i - 1,]);$ 
7      $\text{this} \leftarrow \text{FindCompress}(Im[i, j]);$ 
8     if left compatible with this then //  $O(\text{left}, \text{this}) == 1$ 
9       if  $\text{left} < \text{this}$  then  $\text{Union}(\text{left}, \text{this})$  else  $\text{Union}(\text{this}, \text{left})$ 
10    end
11    if up compatible with this then //  $O(\text{up}, \text{this}) == 1$ 
12      if  $\text{up} < \text{this}$  then  $\text{Union}(\text{up}, \text{this})$  // this is put under
13        up to keep tree as flat as possible
14      else  $\text{Union}(\text{this}, \text{up})$  // this linked to up
15    end
16  foreach element  $e$  of the line  $i$  do  $\text{FindCompress}(p)$ 
17 end

```

算法 13.2: 算法示例 2

第十四章 代码环境

模板中使用 listings 宏包实现代码环境。详细用法见宏包的官方说明文档。

以下是代码示例，可以在文中任意位置引用??。

代码 14.1 示例代码

```
1 #include <stdio.h>
2
3 int main( )
4 {
5     printf("hello, \world\n");
6     return 0;
7 }
```


第十五章 引用文献标注

15.1 著者 -出版年制标注法

<code>\citestyle{ustcauthoryear}</code>	
<code>\cite{knuth86a}</code>	\Rightarrow Knuth (1986)
<code>\citet{knuth86a}</code>	\Rightarrow Knuth (1986)
<code>\citet[chap.~2]{knuth86a}</code>	\Rightarrow Knuth (1986, chap. 2)
<code>\citep{knuth86a}</code>	\Rightarrow (Knuth, 1986)
<code>\citep[chap.~2]{knuth86a}</code>	\Rightarrow (Knuth, 1986, chap. 2)
<code>\citep[see][]{knuth86a}</code>	\Rightarrow (see Knuth, 1986)
<code>\citep[see][chap.~2]{knuth86a}</code>	\Rightarrow (see Knuth, 1986, chap. 2)
<code>\citet*{knuth86a}</code>	\Rightarrow Knuth (1986)
<code>\citep*{knuth86a}</code>	\Rightarrow (Knuth, 1986)
<code>\citet{knuth86a,tlc2}</code>	\Rightarrow Knuth (1986); Mittelbach et al. (2004)
<code>\citep{knuth86a,tlc2}</code>	\Rightarrow (Knuth, 1986; Mittelbach et al., 2004)
<code>\cite{knuth86a, knuth84}</code>	\Rightarrow Knuth (1984, 1986)
<code>\citet{knuth86a, knuth84}</code>	\Rightarrow Knuth (1984, 1986)
<code>\citep{knuth86a, knuth84}</code>	\Rightarrow (Knuth, 1984, 1986)

15.2 顺序编码制标注法

`\citestyle{ustcnumerical}`

<code>\cite{knuth86a}</code>	\Rightarrow	[2]
<code>\citet{knuth86a}</code>	\Rightarrow	Knuth ^[2]
<code>\citet[chap.~2]{knuth86a}</code>	\Rightarrow	Knuth ^[2] , chap. 2]
<code>\citep{knuth86a}</code>	\Rightarrow	[2]
<code>\citep[chap.~2]{knuth86a}</code>	\Rightarrow	[2] chap. 2
<code>\citep[see][]{knuth86a}</code>	\Rightarrow	see ^[2]
<code>\citep[see][chap.~2]{knuth86a}</code>	\Rightarrow	see ^[2] chap. 2
<code>\citet*{knuth86a}</code>	\Rightarrow	Knuth ^[2]
<code>\citep*{knuth86a}</code>	\Rightarrow	[2]
<code>\citet{knuth86a,tlc2}</code>	\Rightarrow	Knuth ^[2] , Mittelbach et al. ^[3]
<code>\citep{knuth86a,tlc2}</code>	\Rightarrow	[2,3]
<code>\cite{knuth86a, knuth84}</code>	\Rightarrow	[1,2]
<code>\citet{knuth86a, knuth84}</code>	\Rightarrow	Knuth ^[1,2]
<code>\citep{knuth86a, knuth84}</code>	\Rightarrow	[1,2]
<code>\cite{knuth86a, knuth84, tlc2}</code>	\Rightarrow	[1–3]

15.3 其他形式的标注

<code>\citealt{tlc2}</code>	\Rightarrow	Mittelbach et al. ³
<code>\citealt*{tlc2}</code>	\Rightarrow	Mittelbach, Goossens, Braams, and Carlisle ³
<code>\citealp{tlc2}</code>	\Rightarrow	³
<code>\citealp*{tlc2}</code>	\Rightarrow	³
<code>\citealp{tlc2, knuth86a}</code>	\Rightarrow	^{2,3}
<code>\citealp[pg.~32]{tlc2}</code>	\Rightarrow	³ pg. 32
<code>\citenum{tlc2}</code>	\Rightarrow	3
<code>\citetext{priv.\ comm.}</code>	\Rightarrow	[priv. comm.]
<code>\citeauthor{tlc2}</code>	\Rightarrow	Mittelbach et al.
<code>\citeauthor*{tlc2}</code>	\Rightarrow	Mittelbach, Goossens, Braams, and Carlisle
<code>\citeyear{tlc2}</code>	\Rightarrow	2004
<code>\citeyearpar{tlc2}</code>	\Rightarrow	2004

chapter/format +=

参考文献

- Knuth D E. May 1984. Literate programming[J]. *The Computer Journal*. 27(2):97–111.
- Knuth D E. 1986. Computers and Typesetting: A The \TeX book[M]. Reading, MA, USA: Addison-Wesley.
- Mittelbach F, Goossens M, Braams J, et al. 2004. The \LaTeX Companion[M]. 2nd ed. Reading, MA, USA: Addison-Wesley.