Mentara-校园心理健康互助社区

软件架构文档

版本 <1.5>

[注：用方括号括起来并以蓝色斜体（样式=InfoBlue）显示的文本，它们用于向作者提供指导，在发布此文档之前应该将其删除。按此样式输入的段落将被自动设置为普通样式（样式=Body Text）。]

[要定制 Microsoft Word 中的自动字段（选中时显示灰色背景），请选择 File>Properties，然后将 Title、Subject 和 Company 等字段替换为此文档的相应信息。关闭该对话框后，通过选择 Edit>Select All（或 Ctrl-A）并按 F9，或只是在字段上单击并按 F9，可以在整个文档中更新自动字段。对于页眉和页脚，这一操作必须单独进行。按 Alt-F9，将在显示字段名称和字段内容之间切换。有关字段处理的详细信息，请参见 Word 帮助。]

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <日/月/年> | <x.x> | <详细信息> | <姓名> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 参考资料 4

2. 用例视图 4

3. 逻辑视图 4

4. 部署视图（可选） 4

5. 进程视图（可选） 4

6. 实现视图（可选） 4

7. 技术视图 4

8. 数据视图（可选） 5

9. 算法视图（可选） 5

10. 性能视图（可选） 5

11. 可靠性视图（可选） 5

12. 安全性视图（可选） 5

13. 易用性视图（可选） 5

14. 可维护性视图（可选） 5

软件架构文档

# 简介

## 目的

本文档将从构架方面对系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面作出的重要决策。

[本节确定此**软件构架文档**在整个项目文档中的作用或目的，并对此文档的结构进行简要说明。应确定此文档的特定读者，并指出他们应该如何使用此文档。]

## 参考资料

[本小节应完整地列出此**软件构架文档**中其他部分所引用的所有文档。每个文档应标有标题、报告号（如果适用）、日期和出版单位。列出可从中获取这些参考资料的来源。这些信息可以通过引用附录或其他文档来提供。]

# 用例视图

[本节列出用例模型中的一些用例或场景，这些用例或场景应体现最终系统中重要的、核心的功能；或是在构架方面涉及范围很广（使用了许多构架元素）；或强调或阐明了构架的某一具体的细微之处。]

# 逻辑视图

[本节说明系统的逻辑结构，例如系统被分解为多个子系统和包。而每个重要的包又被分解为多个类。您应该介绍那些在构架方面具有重要意义的包和类，并说明它们的职责，以及几项非常重要的关系、操作和属性。]

# 部署视图（可选）

[本节说明用来部署和运行该软件的一种或多种物理网络（硬件）配置。对于每种配置，它至少应该指出执行该软件的物理节点（计算机、CPU）及其互连情况（总线连接、LAN 连接、点到点连接等）。另外还要包括**进程视图**中的各进程到物理节点的映射。则本视图本节可不写，在此说明理由。如果系统是单机软件，则本视图本节可不写，在此说明理由。]

# 进程视图（可选）

[本节说明将系统分解为轻量级进程（单个控制线程）和重量级进程（成组的轻量级进程）的情况。本节的内容按照各个通信或交互的进程组来进行组织。说明进程之间的主要通信模式，例如消息传递、中断和会合。如果系统是单进程单线程，或者多进程多线程由所选框架实现，则本视图本节可不写，在此说明理由。]

# 实现视图（可选）

[本节说明实现视图的整体结构，包括安装包及其关系、源代码目录等。如果系统仅一个安装包，则本视图本节可不写，在此说明理由。]

# 技术视图

[技术栈的选型，包括编程语言、开发工具、框架、数据库、中间件等。]

# 数据视图（可选）

[从永久性数据存储方面来对系统进行说明。如果几乎或根本没有永久性数据，或者设计模型与数据模型之间的转换并不重要，那么本节可不写，在此说明理由。]

# 算法视图（可选）

[对系统中的核心算法进行设计。如果没有重要的算法，那么本节可不写。]

# 性能视图（可选）

[对系统的性能进行架构设计，列出所采用的性能设计战术。如果系统的性能不重要或者前面的视图已满足性能要求，那么本节可不写。]

# 可靠性视图（可选）

[对系统的可靠性/可用性进行架构设计，列出所采用的可靠性/可用性设计战术。如果系统的可靠性/可用性不重要或者前面的视图已满足可靠性/可用性要求，那么本节可不写。]

# 安全性视图（可选）

[对系统的安全性进行架构设计，列出所采用的安全性设计战术。如果系统的安全性不重要或者前面的视图已满足性能要求，那么本节可不写。]

# 易用性视图（可选）

[对系统的易用性能进行架构设计，列出所采用的易用性设计战术。如果系统的易用性不重要或者前面的视图已满足性能要求，那么本节可不写。]

# 可维护性视图（可选）

[对系统的可维护性进行架构设计，列出所采用的可维护性设计战术。如果系统的可维护性不重要或者前面的视图已满足可维护性要求，那么本节可不写。]

[如果需要，可增加其他视图。]