

系统分析与设计

设计原则及方法

系统设计

是什么？

分析阶段，确定系统的逻辑模型、功能要求，最终确定一个开发方案。

通常包含什么？

1. 技术选型：核心需求及其解决方案，通常会考虑业界常用解决方案，参考是否有开源项目提供借力。
2. 建模：构建领域模型（通常是业务对象模型，是对客观事物的抽象）
3. 数据库设计：数据存储（关系型／非关系型），规范设计
4. 概要设计（模块划分）：基于功能需求划分；面向对象划分
5. 详细设计：模块的实现（算法），局部结构设计

系统设计

原则

1. 结构稳定性：结构的变动将带来用户界面、数据库、模块等的变动，容易导致项目混乱
2. 可扩展性及可修改性：需求的变动是频繁并且正常的，必须保证结构不变的情况下，可灵活做出调整
3. 健壮性：显而易见的Bug不应存在
4. 标准化原则：符合项目团队的规范

领域模型

是什么？

领域内的概念类或现实世界中对象的抽象；专注于分析问题领域（亦即需求）。

构建步骤

1. 从业务描述中提取名词；
2. 从提取出来的名词中总结业务实体，区分名词中的属性、实体，形成问题域中操作实体的集合；
3. 确定模型之间的关系：数据库ER图；系统设计UML

数据库设计

选择数据存储方式

是否涉及多个实体，并且实体间关系复杂

数据库设计

1. 关系型：符合第三范式
2. 非关系型：模型划分清晰

概要设计

是什么

把需求分析得到的系统扩展用例图转换为软件结构和数据结构；建立的是目标系统的逻辑模型，与编程无关。

具体内容

1. 将一个复杂系统进行模块划分、建立模块的层次结构及调用关系、确定模块间的接口
2. 数据结构设计包括数据特征的描述、确定数据的结构特性。

概要设计

模块与功能

1. 模块：相对独立的多个功能的集合
2. 功能：需求层面定义，最小不可分的需求

为什么需要？

1. 协同开发及流程管理，保证系统的稳定性及后期的可维护性
2. 解耦：降低复杂度，提高系统的可扩展性、健壮性

概要设计

划分原则

1. 职责单一：只处理它职责范围内的事情，并且要完整
2. 独立性&封闭性：模块内高内聚，模块间松耦合；开放出来的接口尽可能少，并且简单
3. 消除重复：建立公用模块，减少冗余
4. 易于理解：职能明确，逻辑清晰

方法

1. 基于功能需求划分；
2. 面向抽象模型划分

概要设计

基于功能划分

是什么

基于面向过程的划分方法，根据功能需求分类的方法进行模块的划分

步骤

1. 分析系统的需求，得出需求列表；
2. 对需求进行归类，并划分出优先级；
3. 根据需求对系统进行模块分析，抽取出核心模块；
4. 将核心模块进行细化扩展，逐层得到各个子模块，完成模块划分。

概要设计

基于功能划分

优点

便于任务划分与管理，系统功能接口的制定在面向过程的思想中可以得到良好体现

缺陷

1. 无法降低模块间的耦合
2. 模块中可能涉及多种数据模型

概要设计

面向抽象模型划分

是什么

基于面向对象的划分方法，不以系统的需求作为模块的划分方法，而是以抽象出系统的数据对象模型的思想对模块进行划分。

步骤

1. 根据系统框架抽象出系统的核心数据模型；
2. 根据核心数据模型将系统功能细化，并将数据模型与视图等剥离，细化数据的流向；
3. 依据数据的流向制定模块和接口，完成模块划分；

概要设计

面向抽象模型划分

优点

1. 降低系统之间的数据耦合度
2. 按照数据模型进行模块的划分，降低每个模块所包含的数据复杂程度，简化数据接口设计
3. 接近人的思维方式对问题进行划分，提高系统的可理解性

缺陷

要求高，耗时长