

第二次作业

一、题目

请结合教材相关内容以及网络资料与课件，在对需求属性特征进行定义的基础上，对网络中任意提供的一个需求管理系统进行分析，或者基于个人对需求工程的理解，请尝试对整个需求管理系统中的实体以及属性进行建模，并对整个系统的开发与实现进行设计。（注：可以采用 ER 图，也可以采用类图或其它模型来表示。）

二、需求工程分析设计

1. 需求开发

（一）需求获取：

参与者：需求分析员和项目经理。

功能：系统应允许用户通过不同方式（如问卷、访谈、会议等）收集需求信息。可以通过创建新需求表单，记录需求名称、描述、类型、优先级、相关用户和项目等信息。

目标：确保需求的多样性和全面性，涵盖所有相关利益方的期望。

（二）需求分析：

参与者：需求分析员执行。

功能：系统应支持分析工具，如需求优先级排序、可行性分析工具。需求分析员可以对需求进行分类、过滤和评估，并与用户进行互动，澄清模糊点。

目标：识别需求之间的依赖关系，评估技术和资源的可行性，确保需求的准确性和可实现性。

（三）需求定义：

参与者：需求分析员与项目经理共同完成。

功能：系统应支持详细文档的上传和版本控制，确保每次需求定义都被记录。提供模板以统一需求文档格式。

目标：形成清晰、详细且无歧义的需求文档，为后续开发提供依据。

2. 需求管理

需求管理确保需求在开发生命周期中的一致性和可控性，主要包括：

（一）需求确认：

参与者：客户、项目经理和需求开发员。

功能：系统应提供审核机制，使相关人员能够对需求文档进行审核和评论，并上传确认书或协议。要求在系统中记录审核意见和最终确认结果。

目标：通过明确的沟通达成共识，确保所有参与者对需求的理解一致，并生成具有法律效力的文档。

（二）需求变更控制：

参与者：需求分析员、开发员和项目经理。

功能：系统应允许用户提交变更请求，并包含变更的理由、影响分析及相关文档的更新。变更请求需经过项目经理的审核，审核结果在系统中记录。

目标：确保需求变更过程透明，防止不必要的变更引入混乱。

（三）需求跟踪：

参与者：项目经理及相关团队成员。

功能：系统应提供需求状态的可视化跟踪，包括需求的各个阶段（已获取、已分析、已定义、已确认、正在开发、已开发）。支持生成报告以便于评估开发进度。

目标：使项目经理能随时了解需求的当前状态，帮助做出及时的决策，确保项目按计划推进。

3. 角色与权限

需求管理系统中主要角色及其权限：

（一）系统管理员：

职责：管理系统用户和权限，包括创建、修改和删除用户，设置用户角色及权限。

功能：能对系统信息进行维护和更新，确保系统安全性。

（二）项目经理：

职责：管理项目整体信息，负责审核需求变更和需求确认。

功能：查看和管理项目下的所有需求，跟踪项目进度，协调整个团队。

（三）需求分析员：

职责：负责需求的获取、分析和定义，确保需求文档的准确性。

功能：可以创建、修改和删除需求，记录需求状态，上传文档。

（四）需求开发员：

职责：根据需求文档进行开发，需确保开发成果符合需求。

功能：查看分配给自己的需求，上传开发成果，并更新需求状态。

4. 功能设计

基于角色及其权限，系统应具备以下功能：

（一）系统信息管理和用户管理：

系统管理员应具备最高权限，能够执行用户的创建、修改、删除操作，维护系统角色和权限设置且应能够批量导入或导出用户数据。

用户权限的分配也应明确，确保不同角色只能访问和操作与其职责相关的功能和数据。

（二）项目信息管理：

系统管理员或项目经理应负责项目的创建、修改、删除，项目中应包含详细的基本信息，如项目编号、名称、描述、起始和结束日期、状态等。

每个项目的团队成员信息，包括项目经理、需求分析员、开发员等，应能被记录并展示，确保团队角色分工明确。系统需支持为每个项目分配不同的负责人，并且在项目变更时能够更新这些信息。

（三）需求获取：

需求分析员或项目经理可通过系统创建新的需求。

系统应支持在需求创建时关联项目，并根据需求的类型或优先级，自动生成需求编号和状态标识。

（四）需求分析与需求定义：

需求分析员和项目经理在获取需求后，应通过系统进行需求的详细分析和定义。系统需要提供版本控制功能，允许对需求进行多次修改，并记录修改的内容、时间和原因。

系统应具备可追溯性，允许查看需求定义过程的全部历史，确保需求定义的完整性和一致性。同时，系统可以支持对需求进行分类标记，方便需求分析员根据需求类型或优先级进行筛选和管理。

（五）需求确认：

系统应设置明确的需求确认流程，需求在定义完成后应通过项目经理和相关利益方（如客户）的确认。

每次需求确认过程应记录详细的时间、参与人员、确认内容等信息，并发送自动通知给相关用户。系统还需支持确认过程的跟踪，确保每个需求都经过完整的审核和确认。

（六）需求开发：

系统应为需求开发员提供开发任务的分配和管理功能。开发员可以在系统中查看分配给自己的需求，确保开发工作基于明确的需求定义进行。

系统应具备开发进度跟踪功能，开发员能够按需更新开发状态，如“正在开发”、“已完成”、“等待测试”等。项目经理和需求分析员可通过系统实时查看开发进度，了解需求实现的状态，确保需求开发按时推进。

（七）需求变更控制：

系统需具备严谨的需求变更控制功能，变更请求应由需求分析员、开发人员或项目经理提交，并通过系统的审批流程进行审核。

系统应自动保存变更记录，任何需求变更都应附带详细的变更历史，包括负责人、变更时间和内容。项目经理可以通过系统查看变更对项目进度和资源分配的影响，以便做出合理的调整。

（八）需求跟踪：

系统应提供可视化的需求状态面板，清晰展示每个需求从创建、分析、定义、确认、开发到完成的全流程状态。项目经理可以通过该面板实时掌握项目整体进展，并根据需求状态动态调整开发计划。

三、实体及属性建模

表 3-1 用户表

属性名	含义	数据类型	长度	是否为主属性	是否外键	约束条件
UserNo	用户工号	INT	10	是		Not null Unsigned
UserName	用户姓名	VARCHAR	50			Not null
UserRole	用户权限	VARCHAR	50			Not null
UserCl	联系方式	VARCHAR	20			null

表 3-2 项目表

属性名	含义	数据类型	长度	是否为主属性	是否外键	约束条件
ProjectNo	项目编号	INT	10	是		Not null Unsigned
ProjectName	项目名	VARCHAR	50			Not null
ProjectDes	项目描述	VARCHAR	50			Not null
ProjectStartDate	开始日期	DATE				null
ProjectEndDate	结束日期	DATE				null
ProjectStatus	项目状态	VARCHAR	50			Not null

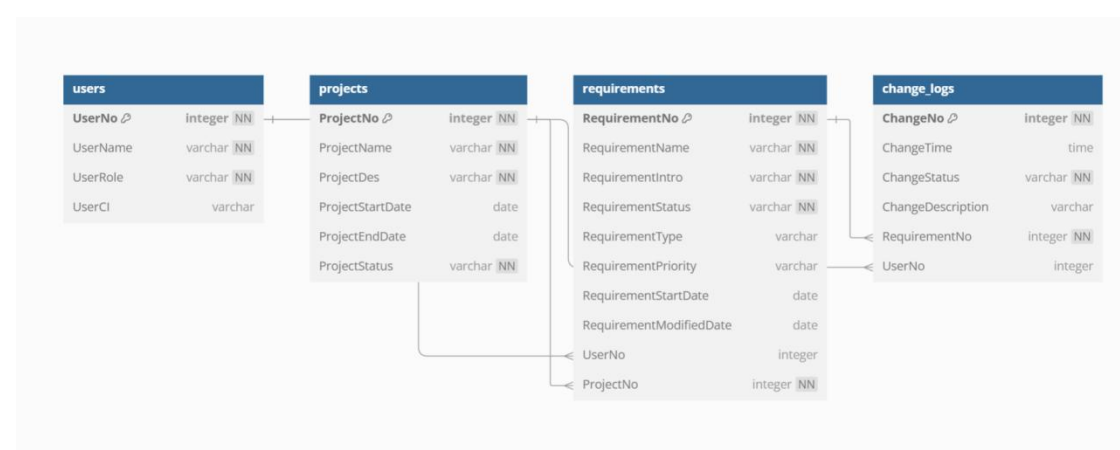
表 3-3 需求表

属性名	含义	数据类型	长度	是否为主属性	是否外键	约束条件
RequirementNo	需求编号	INT	10	是		Not null Unsigned
RequirementName	需求名	VARCHAR	50			Not null
RequirementIntro	需求描述	VARCHAR	50			Not null
RequirementStatus	需求状态	VARCHAR	50			Not null
RequirementType	需求类型	VARCHAR	50			
RequirementPriority	需求优先级	VARCHAR	50			
RequirementStartDate	创建日期	DATE				
RequirementModifiedDate	修改日期	DATE				
UserNo	需求负责人	INT	10		是	
ProjectNo	项目编号	INT	10		是	Not null Unsigned

表 3-4 需求变更记录表

属性名	含义	数据类型	长度	是否为主属性	是否外键	约束条件
ChangeNo	变更编号	INT	10	是		Not null Unsigned
ChangeTime	变更时间	TIME				null
ChangeStatus	变更状态	VARCHAR	50			Not null
ChangeDescription	变更描述	VARCHAR	50			null
RequirementNo	需求编号	INT	10		是	Not null Unsigned
UserNo	负责人	INT	10		是	

对应的 ER 图如下：



附：ER 图由 <https://dbdiagram.io/d> 自动生成，以下是生成 ER 图时使用的代码：

Table users {

UserNo integer [primary key, not null]

UserName varchar [not null]

UserRole varchar [not null]

UserCI varchar

}

Table projects {

ProjectNo integer [primary key, not null]

ProjectName varchar [not null]

ProjectDes varchar [not null]

ProjectStartDate date

ProjectEndDate date

ProjectStatus varchar [not null]

}

Table requirements {

RequirementNo integer [primary key, not null]

RequirementName varchar [not null]

RequirementIntro varchar [not null]

RequirementStatus varchar [not null]

RequirementType varchar

RequirementPriority varchar

RequirementStartDate date

RequirementModifiedDate date

UserNo integer [ref: > users.UserNo]

ProjectNo integer [ref: > projects.ProjectNo, not null]

}

Table change_logs {

ChangeNo integer [primary key, not null]

ChangeTime time

ChangeStatus varchar [not null]

ChangeDescription varchar

RequirementNo integer [ref: > requirements.RequirementNo, not null]

UserNo integer [ref: > users.UserNo]

}