1 功能需求

* 1. 功能划分

本系统的主要任务是对住院部中的各类信息进行查看与管理，主要功能包括登录注册等基本功能，还有人员信息管理功能、病床信息管理功能、科室信息管理功能、病人出入院办理功能、为病人分配护人员及病床功能、消息管理功能和个人信息管理功能，可以让管理员和医护人员便捷的查看自己所需要的信息，并对其进行管理。接下来将对每个功能模块进行简要的需求分析。

1.2 功能描述

（1）用户功能模块：用户功能模块是不涉及具体业务的功能模块，包括登录、注册、修改密码功能和退出登录四种功能。只有医生和护士可以注册账号和修改密码，并且要在注册和修改密码时需要选择自己的身份。当用户登录的时候，也需要选择自己的身份，然后在登陆后要根据用户的身份展示不同的功能模块。用户功能模块的用例图如图1.1所示。

（2）人员信息管理模块：该模块主要功能是对住院部中各类人员基本信息的查看与管理。需要实现对病人、医生和护士的基本信息的展示、查询、修改和删除，还需要对病人的各项生命体征信息进行展示。

（3）病床管理模块：该模块的主要功能是对所有病床信息的总览、对每张病床信息的管理，还需要实现为病人申请与分配病床的功能。只有管理员和医生具有操作这个模块的权限，其中，医生负责为病人进行病床申请，而管理员则负责对病床申请进行审核。

（4）出入院办理模块：该模块的主要功能是对病人的出入院业务进行办理，只有管理员具有访问这一模块的权限。该模块分为包括入院办理和出院办理两个子模块，在为病人办理入院时需要填写病人的基本信息，并且要为其分配主治医生和责任护士；办理出院时需要对相关信息进行更新。

（5）消息管理模块：该模块的主要功能是对自己收到消息的管理。消息分为公告和通知两种类型。所有用户都可以在该模块可以查看自己的收到的消息列表，并可以对某一条消息进行操作，包括查看该消息的详细信息和删除消息；管理员可以在该模块给系统的所有用户发布公告。

（6）个人中心模块：用户可以在该模块查看并修改自己的基本信息。



图1.1 用户功能模块用例图



图1.2管理员用例图

图1.3医生用例图

图1.4护士用例图

2 数据需求

2.1 数据字典

1. 管理员信息表

管理员信息表用来存储管理员的信息。管理员账号密码不能为空，并且密码采用MD5加密算法进加密后将加密后进行保存，管理员的身份默认为0。管理员信息表如表2.1所示。

表2.1管理员息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| id | int | 11 | Primary key/not null | 管理员编号 |
| username | varchar | 256 | null | 姓名 |
| user\_account | varchar | 256 | not null | 登陆账号 |
| user\_password | varchar | 512 | not null | 登陆密码 |
| admin\_gender | varchar | 10 | null | 管理员性别 |
| admin\_age | int | 11 | null | 管理员年龄 |
| phone | varchar | 256 | null | 手机号 |
| user\_role | tinyint | 4 | 0 | 用户身份 |
| avatar\_url | varchar | 1024 | null | 用户头像 |

1. 科室信息表

科室表中存储了住院部所有科室的信息。科室信息表如表2.2所示。

表2.2科室信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| department\_id | int | 11 | Primary key/not null | 科室编号 |
| department\_name | varchar | 256 | not null | 科室名称 |

1. 医生信息表

医生信息表中存储了所有医生的信息，在医生注册时插入数据。医生的账号、密码以及所属科室信息在注册时必须要填写，所以不能为空。医生的密码同样使用了MD5加密算法进行了加密，医生的身份默认为1。医生信息表如表2.3所示。

表2.3医生信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| id | int | 11 | Primary key/not null | 医生编号 |
| username | varchar | 256 | null | 医生姓名 |
| user\_account | varchar | 256 | not null | 登陆账号 |
| user\_password | varchar | 256 | not null | 登陆密码 |
| doctor\_gender | varchar | 10 | null | 医生性别 |
| doctor\_age | int | 11 | null | 医生年龄 |
| phone | varchar | 256 | null | 手机号 |
| avatar\_url | varchar | 1024 | null | 用户头像 |
| department\_id | int | 11 | foreign key /not null | 所属科室编号 |
| user\_role | tinyint | 4 | 1 | 用户身份 |

1. 护士信息表

护士信息表中存储了所有护士的信息，在护士注册时插入数据。其字段的属性同医生信息表类似，护士的身份默认为2。护士信息表如表2.4所示。

表2.4护士信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| id | int | 11 | Primary key/not null | 护士编号 |
| username | varchar | 256 | null | 护士姓名 |
| user\_account | varchar | 256 | not null | 登陆账号 |
| user\_password | varchar | 256 | not null | 登陆密码 |
| nurse\_gender | varchar | 10 | null | 护士性别 |
| nurse\_age | int | 11 | null | 护士年龄 |
| phone | varchar | 256 | null | 手机号 |
| avatar\_url | varchar | 1024 | null | 用户头像 |
| department\_id | int | 11 | foreign key /not null | 所属科室编号 |
| user\_role | tinyint | 4 | 2 | 用户身份 |

1. 病床信息表

病床信息表中存储了各个科室的病床信息，在添加病床时插入数据。病床的所属科室不能为空，在添加新病床时，病床可用状态默认为1表示可用，分配状态默认为0表示未分配。病床信息表如表2.5所示。

表2.5病床信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| bed\_id | int | 11 | Primary key/not null | 病床编号 |
| department\_id | int | 11 | not null | 所属科室编号 |
| available | tinyint | 1 | 1 | 是否可用 |
| allocated | tinyint | 1 | 0 | 是否已被分配 |

1. 病人信息表

病人信息表存储了住院部中所有病人的信息，是系统的关键数据，在为病人办理入院时插入数据。入院日期由后台在插入数据的时候获取当前的日期进行插入。在插入病人信息时，其出院状态默认为0，表示未出院。病人信息表如表2.6所示。

表2.6病人信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| patient\_id | int | 11 | Primary key/not null | 病人编号 |
| patient\_name | varchar | 256 | not null | 病人姓名 |
| patient\_gender | varchar | 10 | null | 病人性别 |
| patient\_age | tinyint | 4 | null | 病人年龄 |
| bed\_id | int | 11 | foreign key /null | 病人病床号 |
| doctor\_id | int | 11 | foreign key /null | 主治医生编号 |
| nurse\_id | int | 11 | foreign key /null | 责任护士编号 |
| is\_discharged | int | 11 | 0 | 病人是否出院 |
| admission\_date | date | 0 | not null | 病人入院日期 |
| discharged\_date | date | 0 | null | 病人出院日期 |
| condition\_description | text | 0 | null | 病情描述 |
| department\_id | int | 11 | foreign key /not null | 所属科室编号 |

1. 病床分配信息表

病床分配信息表中存储了病床分配的信息，用于实现病床的申请分配业务，在医生为病人申请病床的时候插入数据。在插入申请信息时，申请状态默认为2，表示待审核。病床分配信息表如表2.7所示。

表2.7病床分配信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| patient\_id | int | 11 | Primary key/not null | 申请病人编号 |
| apply\_id | int | 11 | not null | 申请者编号 |
| bed\_id | int | 11 | not null | 申请病床编号 |
| department\_id | int | 11 | not null | 所属科室编号 |
| approve\_status | tinyint | 4 | 2 | 申请状态 |

1. 消息表

消息表中存储了所有用户的消息，在用户发送消息时插入数据。在插入消息时，其状态默认为0，表示未读。消息的发送时间在其数据插入数据库时根据当前时间自动生成。消息表如表2.8所示。

表2.8消息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| notice\_id | int | 11 | Primary key/not null | 消息编号 |
| send\_id | int | 11 | not null | 发送者编号 |
| receive\_id | int | 11 | not null | 接收者编号 |
| receive\_role | tinyint | 4 | not null | 接收者身份 |
| avatar | varchar | 1024 | null | 消息图标 |
| title | varchar | 256 | null | 消息标题 |
| content | text | 0 | null | 消息内容 |
| is\_read | tinyint | 4 | 0 | 状态 |
| data\_time | datetime | 0 | CURRENT\_TIME | 发送时间 |
| type | varchar | 32 | not null | 消息类型 |

1. 病人健康信息表

病人健康信息表存储了病人的各项生命体征信息，其数据应与病人信息表对应，在病人办理入院时插入数据。病人健康信息表如表2.9所示。

表2.9病人健康信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| patient\_id | int | 11 | Primary key/not null | 病人编号 |
| patient\_name | varchar | 256 | null | 病人姓名 |
| heart\_rate | int | 11 | null | 心率 |
| pulse | int | 11 | null | 脉搏 |
| temperature | float | 0 | null | 体温 |
| blood\_pressure | float | 0 | null | 血压 |
| breathe | int | 11 | null | 呼吸 |

2.2 数据库描述

管理员（编号，姓名，性别，年龄，手机号，登录账号，登陆密码，用户头像，用户身份）；

科室（编号，科室名称）；

医生（编号，姓名，性别，年龄，手机号，登录账号，登陆密码，用户头像，用户身份，所属科室编号）；

护士（编号，姓名，性别，年龄，手机号，登录账号，登陆密码，用户头像，用户身份，所属科室编号）；

病床（编号，是否可用，是否分配，所属科室编号）；

病人（编号，姓名，性别，年龄，是否出院，入院日期，出院日期，病情描述，所属科室编号，病床号，主治医生编号，责任护士编号）；

病床分配（申请病人编号，申请病床编号，申请者编号，所属科室编号，申请状态）；

消息（编号，接收者身份，消息图标，消息标题，消息内容，发送时间，消息类型，是否已读，发送者编号，接收者编号）。



图2. 系统E-R模型图

3 性能需求

3.1 响应速度

本系统要求对于一些基础的操作响应时间尽量保持在2秒以内，对于一些复杂的操作，也要将响应时间控制在5秒以内。

3.2 可靠性

在系统运行期间，要保证系统不会崩溃，不会发生数据显示混乱、错误甚至丢失，不会出现数据泄漏等严重的安全性问题。

3.3 易用性

要确保系统的界面简洁、结构清楚、导航清晰，各功能模块要设计的合理且易用，以便用户能够流畅的进行各种操作。

3.4 可延展性

为适应未来对各种功能的扩充，系统必须满足二次开发条件，且必须保证系统代码与体系结构之间的低耦合度。

4安全需求

住院部信息管理系统的数据库中，既存储了病人、医生、护士等人员的基本信息，还存储了每个病人详细的病例信息，一旦发生数据泄露等重大安全事故，将对患者造成严重影响，所以系统的安全性与可靠性非常重要。因而在进行系统设计时必须满足以下安全需求：

（1）系统在前端对数据库进行访问时，必须由后端实现有效分隔，不能直接对数据库进行操作；

（2）在后端接收前端传来的数据时，必须要添加必要的校验和过滤，以防有用户发起恶意请求以盗取数据库中的信息；

（3）后端向前端传递数据之前必须要进行脱敏，只将必要信息传回给前端，不能将查询到的整个实体类信息全都传给前端，以防一些等敏感信息的泄露；

（4）用户的密码等敏感信息不能明文存储到数据库中，一定要经过加密之后再进行存储，这样即使数据库信息发生了泄露，也可以在最大程度上减少损失；

（5）对于登录系统的用户一定要进行严格的权限控制，避免非法访问或越权访问的发生。

5 运行需求

5.1 软件接口

（1）操作系统：Windows

（2） 软件设备：本系统基于前后端分离模式进行开发，后端采用Java语言及Spring Boot企业级框架进行开发，前端采用React框架 + Ant Design组件库进行开发，并使用了Ant Design Pro脚手架作为前端项目模板，数据库采用了MySQL关系型数据库。前端用的项目模板：基于Ant Design设计体系的一系列框架如Umi框架、Ant Design Pro脚手架、ProComponents组件库。

5.2 硬件接口

（1）内存：8G

（2）CPU：2.0GHz以上微处理器

（3）硬盘空间：256G