### ****出租房屋管理系统概要设计****

### ****一、引言****

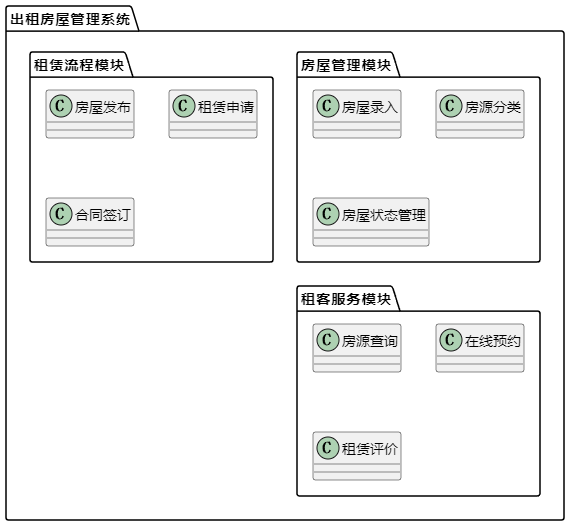
#### ****1. 编写目的****

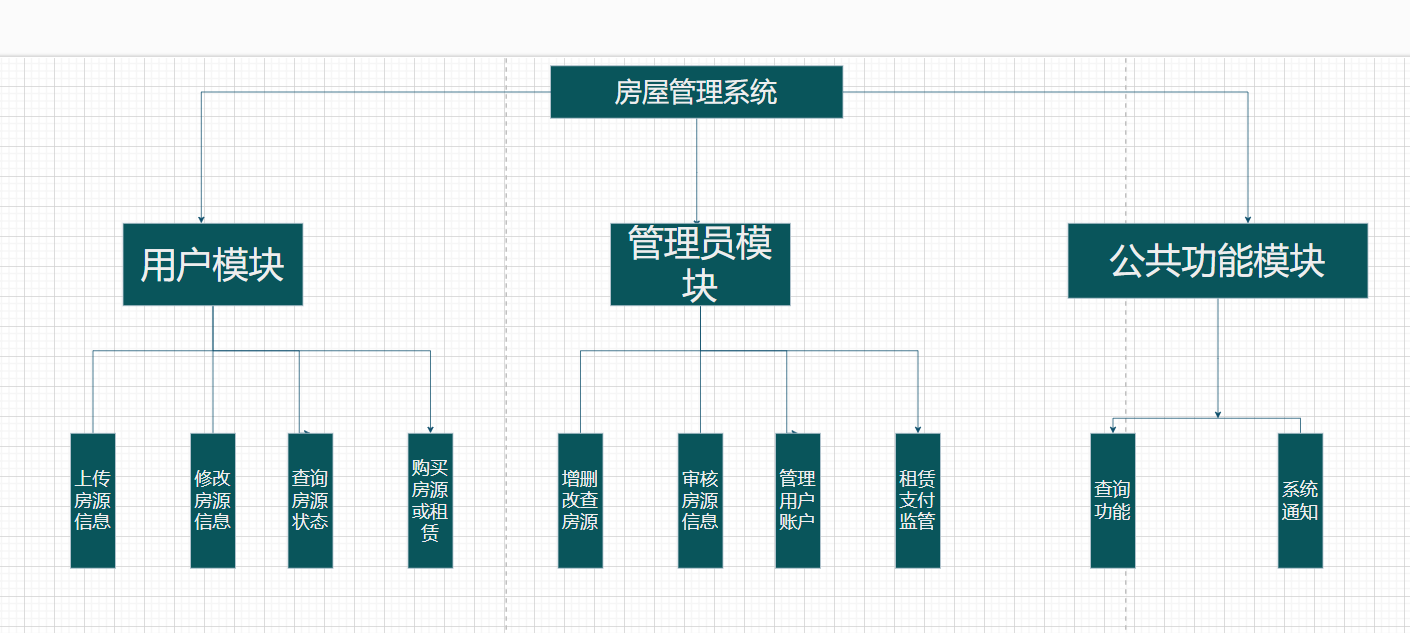
本文档将概要描述出租房屋管理系统的软件功能和软件结构，包括总体设计、接口设计、系统数据结构设计和系统出错处理设计等。该文档的读者包括系统设计人员、软件开发人员和项目评审人员。

#### ****2. 背景****

* **待开发的软件系统名称**：出租房屋管理系统
* **本项目的提出者**：某课程教师
* **开发者**：本课程学生
* **用户**：租客和房东

### ****功能模块****





#### ****1. 房屋管理模块****

* **房屋信息录入**：
  + 支持录入房屋详细信息（如地址、面积、租金、装修情况等），并上传房屋图片。
  + 提供批量导入功能，通过 Excel 文件快速录入多套房源信息。
* **房源分类管理**：
  + 按房源类型（整租、合租、公寓等）分类，支持新增、修改、删除分类。
* **房屋状态管理**：
  + 实时记录房屋状态（空置、已出租、维修中等），并提供状态变更日志。
  + 库存预警功能，提醒房东或管理员更新房源。

#### ****2. 租赁流程模块****

* **房屋发布**：
  + 支持房东发布房源信息，并实时更新状态。
* **租赁申请**：
  + 租客可在线申请租赁房源，提交基本信息，系统进行初步审核后交由房东确认。
* **合同签订**：
  + 提供电子合同模板，支持在线签约，并记录合同详细信息（起租日期、租金、租期等）。

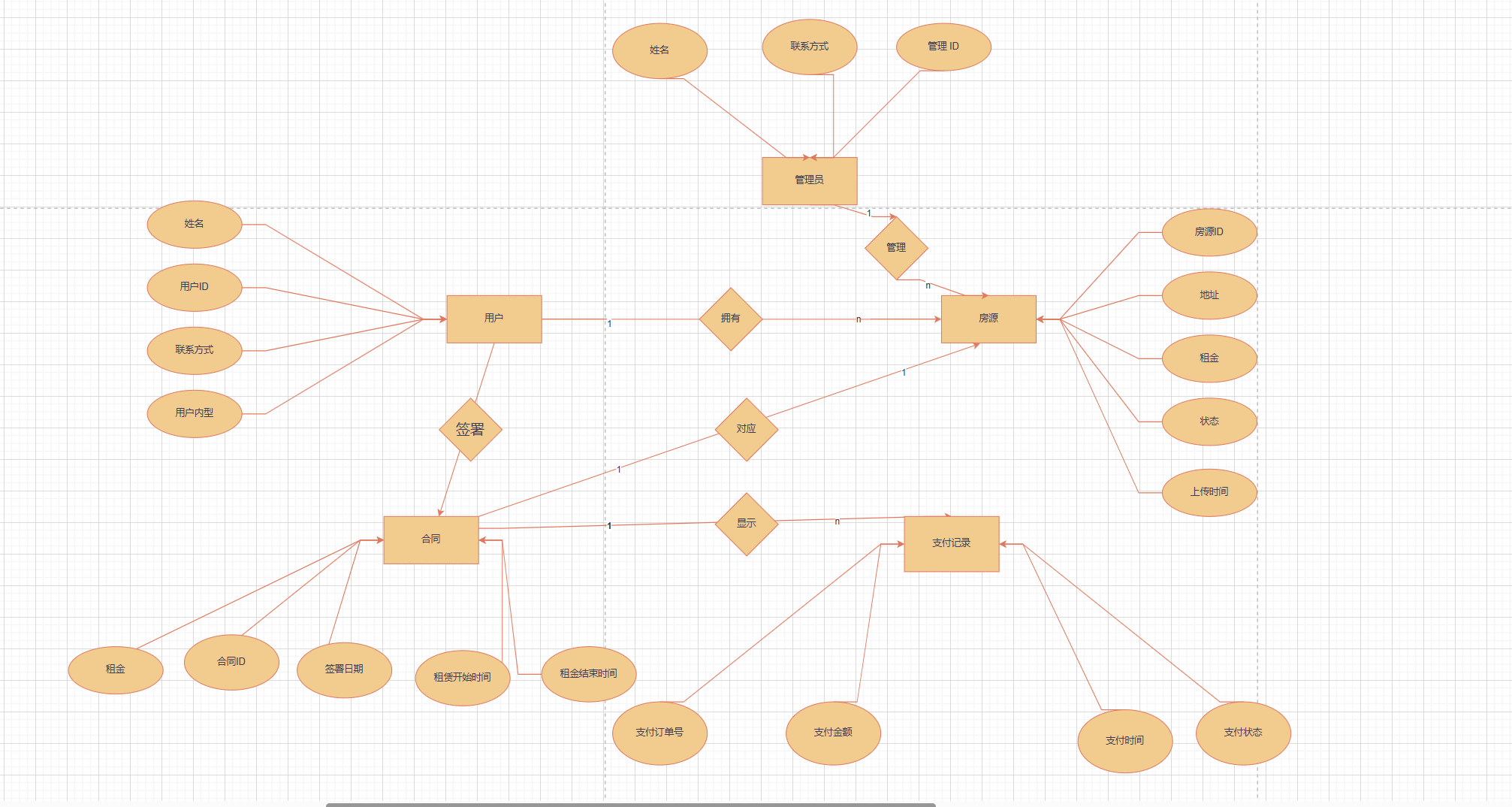
#### ****3. 租客服务模块****

* **房源查询**：
  + 租客可按多种条件（如地段、租金、户型等）查询房源，支持模糊搜索和智能推荐。
* **在线预约**：
  + 租客可在线预约看房，系统将预约信息发送给房东确认。
* **租赁评价**：
  + 租客可以对房屋状况、房东服务等进行评价，为后续租客提供参考。

#### ****4. 系统管理模块****

* **用户账号管理**：
  + 对房东、租客和管理员账号进行细粒度权限划分。
* **参数设置**：
  + 可设置租赁规则（如租赁期限、押金比例等）、系统通知内容等。
* **统计与分析**：
  + 提供租赁报表、租金收入分析等统计数据，辅助房源运营决策。

### ****数据库设计 ER 图****



#### ****1. 实体****

##### ****（1）房源****

* **房源 ID**：主键，唯一标识每套房源。
* **地址**：房屋所在的具体地址。
* **面积**：房屋面积（单位：平方米）。
* **租金**：房屋租金（单位：元）。
* **房屋类型**：如整租、合租等。
* **房屋状态**：空置、已出租、维修中等。
* **房屋图片**：存储房屋图片路径。

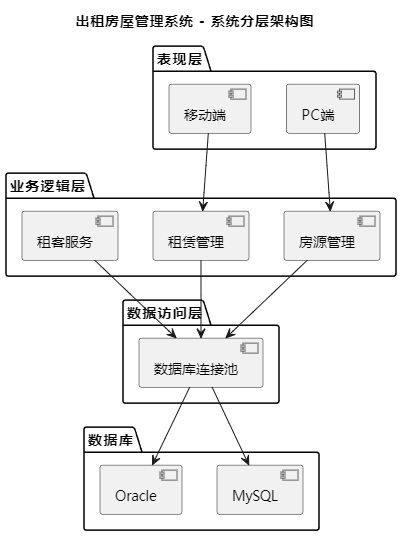
##### ****（2）租客****

* **租客 ID**：主键，唯一标识每位租客。
* **姓名**：租客姓名。
* **联系方式**：手机号码、电子邮箱等。
* **身份证号**：用于实名制管理。
* **注册日期**：租客注册时间。

##### ****（3）租赁记录****

* **租赁 ID**：主键，唯一标识每次租赁行为。
* **房源 ID**：关联租赁的房源。
* **租客 ID**：关联租赁的租客。
* **起租日期**：租赁合同开始日期。
* **到期日期**：租赁合同到期日期。
* **租金**：租赁期间总费用。

### ****系统结构****



#### ****1. 表现层****

* 提供响应式设计，支持 PC 端和移动端访问。
* 用户界面设计简洁直观，房源查询页面提供关键筛选条件，租赁申请页面清晰展示操作步骤。

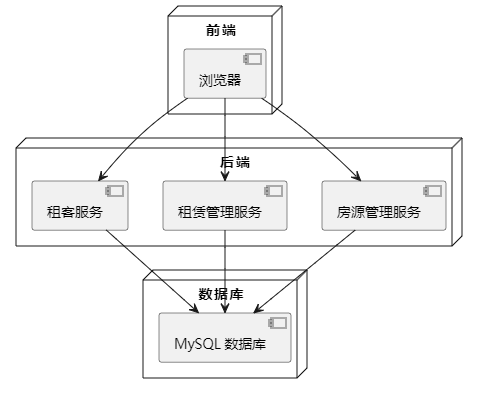
#### ****2. 业务逻辑层****

* **房屋管理业务**：
  + 房屋信息校验（如租金范围、图片格式验证）。
  + 自动更新房源状态（如租赁成功后状态变为“已出租”）。
* **租赁管理业务**：
  + 租赁规则校验（如租期限制、押金计算等）。
  + 自动生成电子合同，并向房东和租客发送通知。
* **统计分析业务**：
  + 根据房源和租赁记录生成报表，如出租率、租金趋势等。

#### ****3. 数据访问层****

* 提供统一的数据访问接口，封装数据库操作逻辑。
* 支持多种数据库类型（如 MySQL、PostgreSQL），方便系统扩展。

### ****运行平台及架构****



#### ****1. 运行平台****

* **服务器端**：
  + 配置：多核处理器、16GB 内存、1TB 硬盘空间。
  + 运行环境：安装 Node.js 或 Java Spring Boot 用于业务逻辑处理。
* **客户端**：
  + 支持主流浏览器（如 Chrome、Firefox）和移动客户端（Android 和 iOS）。

#### ****2. 架构设计****

* 采用 B/S 架构：
  + 前端使用 Vue.js 或 React.js 开发用户界面。
  + 后端通过 RESTful API 提供服务，使用 Spring Boot 实现业务逻辑。
* 若采用微服务架构：
  + 将房屋管理、租赁管理、用户管理等模块拆分为独立的微服务。
  + 使用 Docker 容器化部署，并通过 Kubernetes 管理微服务集群。

### ****六、总结****

出租房屋管理系统旨在通过信息化手段提升租赁服务效率，优化租客与房东的交互体验，并支持房源运营的精细化管理。通过合理的系统设计和实现，该系统将为房屋出租行业提供一个高效、安全、可靠的解决方案。