

智舍宿舍管理系统 — 软件需求规格说明书 (SRS)

Table of Contents

- [▼ 1 Introduction \(简介\)](#)
 - [1.1 背景与动机](#)
 - [1.2 项目目标与范围](#)
 - [1.3 预期用户与受益人](#)
 - [1.4 假设与约束](#)
 - [1.5 文档结构预告](#)
- [▼ 2 Strategic Analysis \(战略分析\)](#)
 - [2.1 SWOT分析](#)
 - [2.2 TOWS策略矩阵](#)
 - [2.3 业务目标与关键举措](#)
- [3 Roadmap \(路线图\)](#)
- [▼ 4 Use Case Modelling & Business Process Modelling \(用例与业务流程\)](#)
 - [4.1 参与者 \(Actor\)](#)
 - [▼ 4.2 用例图 \(Use Case Diagrams\)](#)
 - [4.2.1 整体概览与功能点详解](#)
 - [▼ 4.2.2 子系统用例图与详细用例描述](#)
 - [▼ 用户管理系统](#)
 - [用例名: 登录 \(UC01\)](#)
 - [用例名: 注册 \(UC02\)](#)
 - [▼ 智能分配系统](#)
 - [用例名: 执行智能分配 \(UC03\)](#)
 - [用例名: 获取学生画像 \(UC03A\)](#)
 - [用例名: 可视化选房 \(UC04\)](#)
 - [用例名: 筛选宿舍条件 \(UC05\)](#)
 - [用例名: 提交调宿申请 \(UC06A\)](#)
 - [用例名: 管理调宿申请 \(UC06B\)](#)
 - [用例名: 审批调宿申请 \(UC06C\)](#)
 - [▼ 维修系统](#)
 - [用例名: 提交报修 \(UC07\)](#)
 - [用例名: 智能派单 \(UC08\)](#)
 - [用例名: 接收工单 \(UC09A\)](#)
 - [用例名: 更新维修状态 \(UC09B\)](#)
 - [用例名: 追踪维修进度 \(UC10\)](#)
 - [用例名: 评价维修服务 \(UC11\)](#)
 - [用例名: 监控维修闭环 \(UC12\)](#)
 - [▼ 缴费系统](#)
 - [用例名: 自动计费 \(UC13\)](#)
 - [用例名: 读取水电数据 \(UC14\)](#)
 - [用例名: 在线支付 \(UC15\)](#)
 - [用例名: 调用支付接口 \(UC15A\)](#)
 - [用例名: 管理账单 \(UC16A\)](#)
 - [用例名: 接收欠费预警 \(UC16B\)](#)
 - [▼ 访客系统](#)
 - [用例名: 提交访客预约 \(UC17A\)](#)
 - [用例名: 处理访客登记 \(UC17B\)](#)

- [用例名: 使用门禁 \(UC18A\)](#)
- [用例名: \(安保\)处理认证 \(UC18B\)](#)
- [用例名: 请求身份验证 \(UC18C\)](#)
- [用例名: 响应异常预警 \(UC19\)](#)
- [用例名: 管理安防报表 \(UC20\)](#)
- [4.3 活动图](#)
 - [宿舍智能分配](#)
 - [设备报修](#)
 - [调宿申请管理](#)
 - [访客管理](#)
 - [在线缴费](#)
- [5 Glossary of Terms \(术语表\)](#)
- [6 Supplementary Specification \(补充规格\)](#)
- [7 UI Mock-ups \(界面快照\)](#)
 - [7.1 登录页面 \(Login Page\)](#)
 - [7.2 个人宿舍主页 \(Student Dorm Dashboard\)](#)
 - [7.3 报修申请页面 \(Repairment Request Page\)](#)
 - [7.4 宿舍管理员主界面 \(Manager Dashboard\)](#)
 - [7.5 易用性与无障碍性说明](#)
 - [7.6 章节总结](#)
- [8 AI Tools Usage Acknowledgement \(ai工具使用声明\)](#)
- [9 References \(参考文献\)](#)
- [10 Contributions of Team Members \(成员贡献\)](#)

1 Introduction (简介)

1.1 背景与动机

当前高校宿舍管理普遍存在系统割裂、信息孤立和人工处理效率低下等问题。宿舍分配、维修报修、费用统计与安防管理等模块往往分散在不同的平台中，缺乏统一的数据接口与集中管理机制。宿管人员需要在多个系统间切换操作，学生难以及时了解报修进度或宿舍变动情况，管理层也难以获得全局数据用于决策支持。此外，宿舍能源与安全数据尚未被充分利用，导致潜在的分析与优化价值被浪费。在这种背景下，构建一个覆盖宿舍全生命周期、集成化与智能化的宿舍管理系统具有重要意义。它不仅能提升宿舍管理的自动化与透明度，还能通过数据整合与可视化分析支持学校的精细化管理。

1.2 项目目标与范围

本项目旨在开发一个 **基于 Web 平台的高校宿舍综合管理系统**，实现宿舍业务的数字化与高效化。系统的设计目标是整合宿舍相关的核心场景，具体包括：

- **宿舍分配管理**：支持学生入住登记、宿舍分配与调换；
- **报修管理**：提供报修申请、任务流转与维修反馈；
- **费用管理**：记录宿舍水电及物业费用，支持统计与缴费追踪；
- **安防管理**：对接门禁系统与安全日志，实现进出记录与异常提示；
- **数据分析与可视化**：为校方管理者提供统计报表与趋势分析。

系统采用模块化架构，支持功能扩展，例如访客管理、宿舍考核及物资申领等子系统。

1.3 预期用户与受益人

该系统面向高校宿舍管理相关的多角色用户群体：

- 学生用户**：提交报修、查看费用与宿舍信息；
- 宿管人员**：进行宿舍分配、公告发布与信息维护；
- 维修人员**：接收任务、反馈进度并更新维修状态；
- 安保人员**：监控门禁数据与安防日志；
- 学校管理层**：通过数据分析模块进行决策与资源调配；
- 访客用户 (扩展场景)**：可通过登记功能完成访问申请。

通过统一系统，各角色间的信息交互将更加顺畅，减少沟通成本，提升整体管理效率与用户体验。

1.4 假设与约束

系统在设计与实现过程中基于以下假设与约束条件：

- 运行平台**：主要部署于 Web 端，并支持通过 Electron 封装实现跨平台桌面应用；
- 技术栈选型**：前端采用 TypeScript + React，后端基于 Go 语言与 Gin 框架开发，数据库使用 PostgreSQL；
- 数据合规与隐私保护**：系统需符合高校信息安全要求，所有用户数据需加密存储与访问控制，确保不同权限间的信息隔离；
- 部署环境约束**：系统预计运行在校内服务器或私有云环境中，需支持多终端访问与局域网优化。

1.5 文档结构预告

本报告后续章节将继续阐述项目的战略分析（第 2 章）、用例建模与流程设计（第 4 章）、非功能性需求指标（第 6 章）及用户界面说明（第 7 章）。通过完整的系统分析与设计说明，本文档旨在为宿舍管理系统的开发与实现提供清晰的参考与标准化依据。

2 Strategic Analysis (战略分析)

2.1 SWOT分析

优势 (Strengths)

- 团队技术扎实**：成员熟悉现代技术栈 (TypeScript + React + Go + PostgreSQL)，具备自主开发与迭代能力。
- 校内场景理解深刻**：作为在校学生，对宿舍管理的实际痛点（如分配不透明、报修流程繁琐）有切身体验。
- 同济内部资源可用性**：可借助同济大学国家大学科技园体系建设方案中“校内场景验证项目”的机会，将系统作为校园数字化成果进行试点，并获取校内数据与政策支持。
- 模块化架构**：系统支持功能扩展（如访客管理、能耗分析），易于后续结合校内需求增加模块。

劣势 (Weaknesses)

- 项目规模与经验不足**：相较于成熟商业系统（如“智慧校园”类产品），团队在项目部署、运维经验上存在不足。
- 初始数据整合难度**：宿舍、学籍等历史数据需从学校现有系统迁移，数据清洗与接口对接工作量较大。
- 资金与资源限制**：学生团队在服务器资源、市场推广方面预算有限，可能影响系统初期部署与推广速度。
- 用户基数与接受度**：系统上线后需说服宿管、学生等多角色使用，改变原有工作习惯存在阻力。

机会 (Opportunities)

- **智慧校园建设导向**：同济大学近年来推动“科技成果转化”与“创新创业人才培养”，本项目高度契合学校战略，易获得校内资源支持。
- **市场需求明确**：高校宿舍管理普遍存在系统割裂、效率低下问题，一体化系统具备广泛需求基础。
- **AI与IoT技术成熟**：可引入校内已有的智能门禁、水电表数据，低成本实现安防与能耗管理等扩展功能。
- **环同济知识经济圈生态**：同济与杨浦区共建“环同济知识经济圈”，2023年总产出已超700亿元，区域内企业对校园创新项目有合作与孵化意愿。

威胁 (Threats)

- **竞品压力**：已有成熟的商业宿舍管理系统（如部分高校采用的“酒店式管理系统”）占据市场，部分功能重合。
- **数据安全与合规风险**：系统涉及学生个人信息，需严格遵守《网络安全法》和校内数据管理规定，合规成本高。
- **预算与采购周期**：高校信息化项目采购流程长，若无法纳入学校正式预算，部署进度可能受阻。
- **技术依赖风险**：系统依赖学籍系统、水电表系统等外部接口，接口不稳定可能影响核心功能。

2.2 TOWS策略矩阵

策略类型	策略方向	具体举措
SO (优势-机会)	打造校内标杆案例，形成可复制模式	1. 申请成为同济“校内场景验证项目”，在1-2栋宿舍楼试点，积累运行数据与用户反馈。 2. 与学校后勤部门合作，将系统纳入“智慧宿舍”官方建设计划，争取资源倾斜。
WO (劣势-机会)	借助政策与生态资源弥补初期短板	1. 利用同济“职务科技成果赋权试点”，争取技术成果转化支持，缩短开发与落地周期。 2. 通过“环同济知识经济圈”对接区域内企业，获取硬件（如门禁设备）或云资源赞助。
ST (优势-威胁)	构建差异化能力应对竞争	1. 突出“学生团队开发”的贴近性优势，针对同济校内流程做深度定制，形成商业系统难以复制的场景适配。 2. 与学校技术转移中心合作，将系统数据安全方案标准化，转化为可对外输出的解决方案。
WT (劣势-威胁)	控制风险并确保项目生存	1. 采用分阶段上线策略（参考本项目路线图），优先落地核心功能（MVP1），降低初期复杂度和部署风险。 2. 明确数据边界，所有敏感数据采集与处理均遵循校内规范，避免合规争议。

2.3 业务目标与关键举措

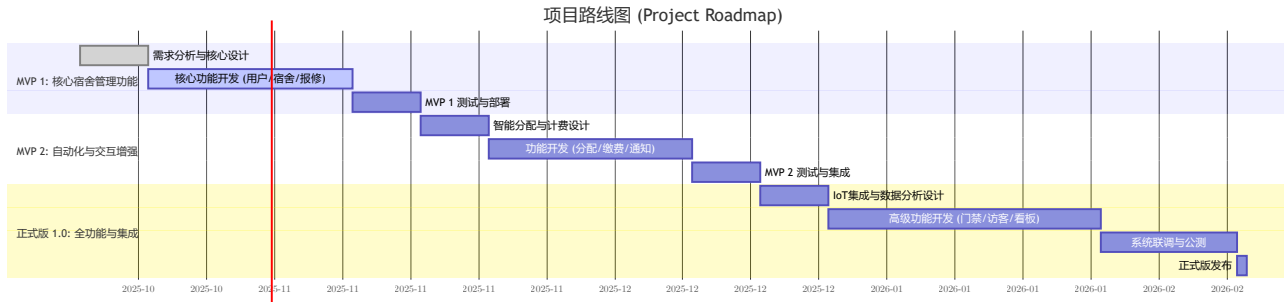
业务目标	关键绩效指标 (KPI)	关键举措	时间节点
成为同济校内智慧宿舍标杆	校内2栋楼试点部署 用户满意度 ≥ 4.3/5	1. 对接学校后勤部，纳入官方试点计划。 2. 完成与学籍系统、门禁系统基础接口对接。	2025年12月
实现核心管理效率提升	报修处理SLA ≤ 24小时 宿舍分配耗时降低50%	1. 上线智能派单与分配算法（MVP2）。 2. 为宿管人员提供专项培训与操作手册。	2026年3月
形成可推广的产品化方案	至少1项功能模块完成产品化封装	1. 将“能耗监测”或“访客管理”模块打包为独立解决方案。 2. 通过同济技术转移网络向外部高校推广。	2026年6月

3 Roadmap (路线图)

本项目采用敏捷迭代的开发模式，将整体目标分解为三个主要阶段：两个最小可行产品（MVP）版本和一个功能完善的正式版。每个阶段都有明确的核心能力、交付成果和验收标准，确保项目方向清晰、风险可控。

- **MVP 1: 核心宿舍管理闭环**
 - **核心能力**: 跑通“学生-管理员”在宿舍管理中的核心业务流程。

- **交付物:**
 - 学生端: 支持登录、查看个人与宿舍信息、提交维修请求。
 - 管理端: 支持学生信息管理、宿舍房间管理、维修工单查看与分配。
- **验收标准:** 学生能成功提交报修, 管理员能在后台看到工单并处理; 基础数据 (学生、宿舍) 可被正确增删改查。
- **MVP 2: 自动化与服务扩展**
 - **核心能力:** 引入智能化与自动化工具, 提升管理效率与用户体验。
 - **交付物:**
 - 智能分配模块: 基于学生画像 (如作息、专业) 推荐宿舍分配方案。
 - 费用管理模块: 支持在线查看水电账单与在线支付。
 - 通知与反馈系统: 支持管理员发布公告、学生查看维修进度与评价。
 - **验收标准:** 系统可根据预设规则自动生成宿舍分配建议; 学生可完成一次完整的在线缴费流程; 维修状态变更后, 学生能收到实时更新。
- **正式版 1.0: 全功能智慧宿舍平台**
 - **核心能力:** 集成物联网与数据分析, 打造一站式智慧宿舍解决方案。
 - **交付物:**
 - IoT集成: 对接智能门禁, 实现刷脸/刷卡出入与异常记录。
 - 访客管理系统: 支持访客在线预约、审批与临时凭证生成。
 - 数据驾驶舱: 为管理者提供宿舍入住率、维修效率、能耗趋势等可视化分析图表。
 - **验收标准:** 授权用户可通过人脸识别进入宿舍; 访客可通过预约生成的二维码在规定时间内进入; 数据驾驶舱能正确反映各项核心指标。



4 Use Case Modelling & Business Process Modelling (用例与业务流程)

4.1 参与者 (Actor)

主要参与者 (Primary Actors):

- 学生 (Student): 宿舍的主要使用者, 系统服务的主要对象。
- 宿管 (Housing Administrator): 负责宿舍分配、审批、费用和维护流程的日常管理。
- 维修人员 (Maintenance Staff): 负责接收工单并执行具体的维修任务。
- 安保 (Security Staff): 负责门禁、访客和异常事件的管理。
- 访客 (Visitor): 需要进入宿舍楼的外部人员。
- 系统管理员 (System Administrator): 负责系统的后台配置、账户管理和规则设定。

外部系统/辅助参与者 (External/Secondary Actors):

- 学籍系统 (Student Records System): 提供学生画像数据。
- 水电表系统 (Utility Meter System): 提供实时的水电用量数据。
- 支付渠道 (Payment Gateway): 处理在线支付请求。

4.2 用例图 (Use Case Diagrams)

4.2.1 整体概览与功能点详解

- 学生是系统的主要用户，功能涵盖住宿、报修、缴费和安防。
- 登录: 学生使用账号密码进入系统。
- 注册: 学生首次使用时创建自己的系统账号。
- 选择房间: 学生在可选范围内直观地挑选自己的宿舍房间。
- 筛选宿舍条件: 学生根据个人习惯（如作息、专业）设置住宿偏好。
- 提交调宿申请: 学生因故需要更换宿舍时，向宿管发起线上申请。
- 提交报修: 学生发现设施损坏时，通过系统快速提交维修请求。
- 追踪维修进度: 学生实时查看自己提交的报修单目前正处于哪个处理阶段。
- 在线支付: 学生通过系统缴纳水电费、网费等宿舍相关费用。
- 接收欠费预警: 当学生账单逾期未缴时，系统会自动发送提醒通知。
- 管理账单: 学生查询自己的历史缴费记录和待缴账单详情。
- 提交访客预约: 学生为即将到访的亲友进行线上预约登记。
- 处理认证: 学生通过刷卡或人脸识别等方式通过宿舍门禁。

6 / 36

- 宿管是系统的核心管理人员，负责分配、运维、账单和安防的全局管理。
- 登录: 宿管使用其管理员账号进入系统后台。
- 注册: 宿管首次使用时创建自己的管理员账号。
- 执行智能分配: 宿管启动系统，利用算法为（例如新生）批量匹配和安排宿舍。
- 管理调宿申请: 宿管查看和管理所有学生提交的调宿申请列表。
- 审批调宿申请: 宿管对学生的调宿申请进行“批准”或“驳回”的操作。
- 监控维修闭环: 宿管在后台查看所有维修工单的状态和维修人员的效率。
- 管理账单: 宿管查看所有宿舍的缴费状态，并可进行（例如自动计费的）管理。
- 处理访客登记: 宿管审批或管理学生提交的访客预约申请。
- 管理安防报表: 宿管查看和导出宿舍楼的出入记录、异常告警等统计报告。

维修人员相关功能

- 维修人员是维修闭环的执行者，负责处理具体的维修任务。
- 登录: 维修人员使用其工号账号登录系统（通常是移动端）。
- 注册: 维修人员首次使用时创建自己的工勤账号。
- 接收工单: 维修人员接收由系统（或宿管）派发的维修任务。
- 更新维修状态: 维修人员在维修的不同阶段（如出发、维修中、已完成）更新工单进度。

安保相关功能

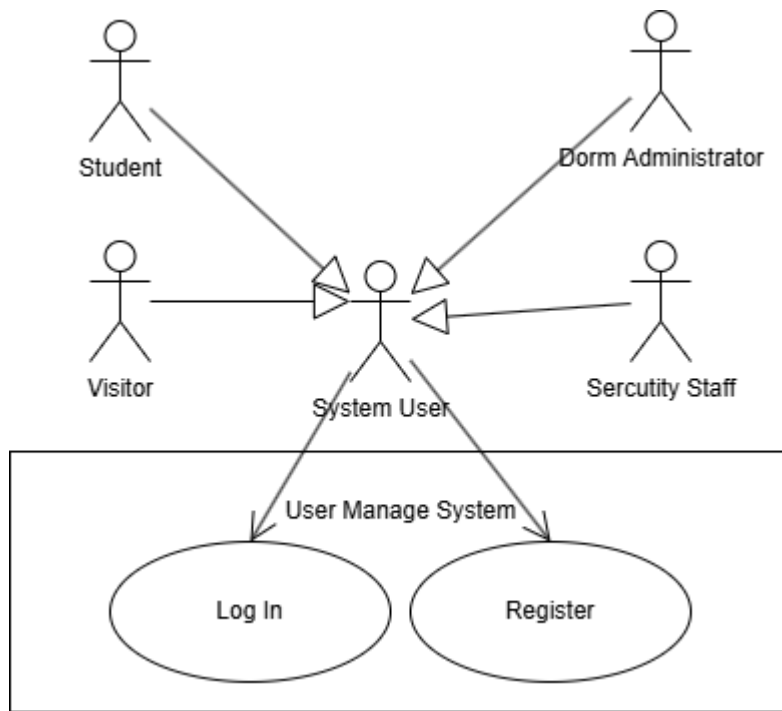
- 安保是宿舍安全的负责人，管理门禁、访客和异常事件。
- 登录: 安保人员使用其账号登录安防管理后台。
- 注册: 安保人员首次使用时创建自己的安防账号。
- 处理访客登记: 安保人员在门岗处核验或审批访客的预约信息。
- 处理认证: 安保人员管理门禁系统，处理学生或访客的出入认证流程。
- 响应异常预警: 当系统检测到如尾随、强行闯入等异常时，安保人员会收到告警并前去处理。
- 管理安防报表: 安保人员查看和导出宿舍楼的出入记录、异常事件等统计报告。

访客相关功能

- 访客是系统的外部人员，仅能在授权后进入宿舍。
- 处理认证: 访客在预约被批准后，使用获得的临时凭证（如二维码）通过宿舍门禁。

4.2.2 子系统用例图与详细用例描述

用户管理系统

**用例名: 登录 (UC01)**

属性	描述
用例名	登录
编号	UC01
参与者	学生, 宿管, 维修人员, 安保 (及其他所有需认证的系统用户)
前置条件	1. 参与者已拥有一个系统账号 (已执行过 UC02)。 2. 参与者已打开系统登录界面。
事件流	1. 参与者输入用户名 (或学号/工号)。 2. 参与者输入密码。 3. 参与者点击“登录”按钮。 4. 系统验证凭据是否正确。 5. 系统根据参与者角色 (学生、宿管等) 授予相应权限并创建会话。 6. 系统跳转到该角色的功能主页。
后置条件	1. 参与者成功登录系统, 可以访问其权限范围内的功能。
扩展场景	4a. 用户名或密码错误。 4a1. 系统在登录页面提示“用户名或密码不正确”。 4a2. 参与者可选择重新输入, 用例结束。

用例名: 注册 (UC02)

属性	描述
用例名	注册
编号	UC02
参与者	学生, 宿管, 维修人员, 安保, (或 系统管理员)
前置条件	1. 参与者尚未拥有系统账号。

属性	描述
事件流	<p>[场景A: 内部用户批量注册 - 主要流程]</p> <ol style="list-style-type: none"> 1. 系统管理员 (参与者) 登录后台管理系统。 2. 系统管理员 访问“用户管理”模块。 3. 系统管理员 通过批量导入 (如Excel) 或与学籍/人事系统对接的方式, 为 学生, 宿管, 维修人员, 安保 批量创建账号并分配初始密码和角色。 <p>[场景B: 临时用户自主注册 - 次要流程]</p> <ol style="list-style-type: none"> 1. 临时用户 (如 访客) 进入注册页面。 2. 用户填写注册信息 (如手机号、姓名、设置密码) 。 3. 系统创建账号, 默认为权限受限的“访客”或“待审核”角色。
后置条件	1. 系统中为新用户创建了账号信息和角色权限。

智能分配系统



用例名: 执行智能分配 (UC03)

属性	描述
用例名	执行智能分配
编号	UC03
参与者	宿管
前置条件	<ol style="list-style-type: none"> 1. 宿管已登录系统 (UC01)。 2. 待分配的学生数据已准备好。 3. 宿舍房源数据已准备好。

属性	描述
事件流	<ol style="list-style-type: none"> 宿管进入“智能分配”模块，选择要分配的学生批次。 宿管设置分配规则（如“同专业优先”、“尊重偏好”）。 宿管启动分配。 系统自动包含并执行 UC03A 获取学生画像。 系统根据算法和学生画像数据，运行匹配程序。 系统生成分配预案并呈现给宿管。 宿管审查并（可选）微调预案。 宿管确认并执行分配。
后置条件	1. 该批次学生被分配到宿舍，数据更新。

用例名: 获取学生画像 (UC03A)

属性	描述
用例名	获取学生画像
编号	UC03A
参与者	系统 (由 UC03 自动触发), 学籍系统
前置条件	<ol style="list-style-type: none"> UC03 执行智能分配 被触发。 学籍系统 接口可用。
事件流	<ol style="list-style-type: none"> 系统接收到 UC03 的数据请求。 系统调用 学籍系统 的接口，请求目标批次学生的画像数据（专业、生源地、偏好等）。 学籍系统 返回数据。 系统将数据提供给 UC03 的分配算法使用。
后置条件	1. UC03 的分配算法获取了所需的数据。

用例名: 可视化选房 (UC04)

属性	描述
用例名	可视化选房
编号	UC04
参与者	学生
前置条件	<ol style="list-style-type: none"> 学生已登录系统 (UC01)。 系统已开放“自主选房”或“调宿选房”功能。
事件流	<ol style="list-style-type: none"> 学生进入“可视化选房”界面。 系统显示学生可选楼栋的楼层平面图和床位状态（如“可选”、“已占用”）。 学生点击一个“可选”床位。 系统显示该床位及同宿舍已有成员的（脱敏）信息。 学生确认选择该床位。 系统锁定该床位，更新学生宿舍信息。
后置条件	<ol style="list-style-type: none"> 学生成功预定或分配到新床位。 宿舍房源信息更新。
扩展场景	<ol style="list-style-type: none"> 在学生确认的瞬间，床位被他人抢占。 <ol style="list-style-type: none"> 系统提示“手慢了，床位已被占用”，要求学生重新选择。

用例名: 筛选宿舍条件 (UC05)

属性	描述
用例名	筛选宿舍条件
编号	UC05
参与者	学生
前置条件	1. 学生已登录系统 (UC01)。 2. 学生处于“智能分配”信息采集阶段, 或 UC04 选房时。
事件流	1. 学生进入“住宿偏好设置”或“条件筛选”页面。 2. 学生设置多维度条件 (如: 作息时间"早睡"、"晚睡"; 兴趣爱好"运动"、"阅读"; 专业"同专业优先"等)。 3. 学生保存偏好。 4. (在 UC04 中) 系统根据条件高亮显示符合偏好的宿舍。 5. (在 UC03 中) 系统将此偏好作为分配算法的输入。
后置条件	1. 学生的住宿偏好被保存到其个人画像中。

用例名: 提交调宿申请 (UC06A)

属性	描述
用例名	提交调宿申请
编号	UC06A
参与者	学生
前置条件	1. 学生已登录系统 (UC01)。 2. 学生当前已有分配好的宿舍。 3. 系统处于允许调宿的时间窗口。
事件流	1. 学生进入“我的宿舍”并选择“调宿申请”。 2. 学生填写申请表格 (如: 调宿理由、目标宿舍偏好等)。 3. 学生确认并提交申请。 4. 系统创建一张“待审批”的调宿工单, 并通知宿管。
后置条件	1. 系统中生成一张待审批的调宿工单。

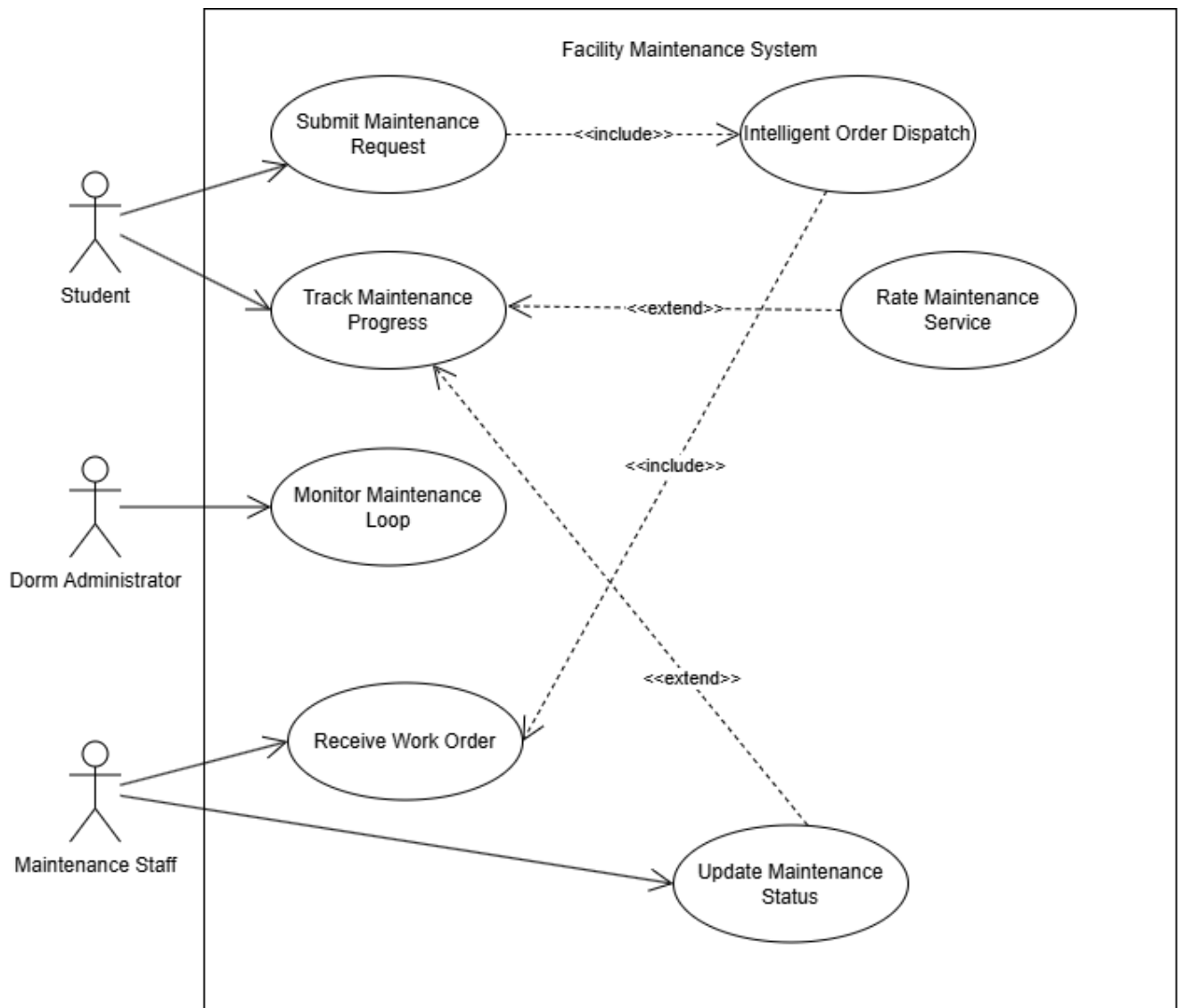
用例名: 管理调宿申请 (UC06B)

属性	描述
用例名	管理调宿申请
编号	UC06B
参与者	宿管
前置条件	1. 宿管已登录系统 (UC01)。 2. 至少有一条由 UC06A 提交的待审批申请。
事件流	1. 宿管进入“调宿审批”列表页面。 2. 系统显示所有“待审批”的申请。 3. 宿管选择一条申请查看详情 (学生理由、当前宿舍、目标宿舍)。 4. 宿管决定处理该申请, 系统执行 UC06C 审批调宿申请。
后置条件	1. 调宿申请的状态被更新 (批准或驳回)。

用例名: 审批调宿申请 (UC06C)

属性	描述
用例名	审批调宿申请
编号	UC06C
参与者	宿管 (由 UC06B 自动触发)
前置条件	1. 宿管在 UC06B 中选择了一条申请进行处理。
事件流	1. 系统显示审批操作界面 (批准/驳回按钮)。 2. [场景A: 批准] 2a1. 宿管点击“批准”。 2a2. 宿管为学生指定一个新床位 (或系统自动推荐)。 2a3. 宿管确认。 2a4. 系统更新学生宿舍信息, 释放旧床位。 2a5. 系统通知学生“申请已批准”。 3. [场景B: 驳回] 3a1. 宿管点击“驳回”。 3a2. 宿管填写驳回理由 (必填)。 3a3. 宿管确认。 3a4. 系统通知学生“申请已驳回” (附带理由)。
后置条件	1. 该调宿申请工单被关闭。

维修系统



用例名: 提交报修 (UC07)

属性	描述
用例名	提交报修
编号	UC07
参与者	学生
前置条件	1. 学生已登录系统。 2. 学生发现宿舍设施有损坏。
事件流	1. 学生进入“提交报修”页面。 2. 学生填写报修信息（如位置、问题描述、上传图片）。 3. 学生确认并提交。
后置条件	1. 系统创建一张新工单。 2. 工单已进入派单流程。

用例名: 智能派单 (UC08)

属性	描述
用例名	智能派单
编号	UC08
参与者	系统 (由 UC07 自动触发)
前置条件	1. UC07 提交报修 被触发。 2. 系统中存在可用的维修人员及其技能/位置信息。
事件流	1. 系统接收到 UC07 的新工单。 2. 系统根据工单的紧急程度、位置和所需技能, 自动匹配最佳的维修人员。 3. 系统将工单推送给该维修人员。
后置条件	1. 工单状态更新为“待接单”。 2. 目标维修人员收到新工单通知。

用例名: 接收工单 (UC09A)

属性	描述
用例名	接收工单
编号	UC09A
参与者	维修人员
前置条件	1. 维修人员已登录系统。 2. UC08 智能派单 已向其推送新工单。
事件流	1. 维修人员收到新工单通知。 2. 维修人员查看工单详情（位置、问题）。 3. 维修人员点击“接单”或“拒单”。 4. [扩展点: 更新状态] 接单后, 维修人员可以在工单处理流程中, 选择 UC09B 更新维修状态。
后置条件	1. 工单状态更新为“维修中” (如果接单)。 2. 工单状态更新为“待重新派单” (如果拒单)。

用例名: 更新维修状态 (UC09B)

属性	描述
用例名	更新维修状态

属性	描述
编号	UC09B
参与者	维修人员
前置条件	1. UC09A 接收工单 处于“维修中”状态。 2. 维修人员在 UC09A 的扩展点选择更新状态。
事件流	1. [$\langle \rangle$] 本用例扩展 UC09A 接收工单。 2. 维修人员在 UC09A 的工单详情页点击“更新状态”。 3. 维修人员选择新状态（如“已出发”、“已完成”）。 4. 若选择“已完成”，维修人员填写维修报告/使用零件。 5. 系统保存状态。
后置条件	1. 工单状态被实时更新。 2. 当状态为“已完成”时，触发 UC10 中的评价扩展点。

用例名: 追踪维修进度 (UC10)

属性	描述
用例名	追踪维修进度
编号	UC10
参与者	学生
前置条件	1. 学生已登录系统。 2. 学生已至少提交过一张报修单 (UC07)。
事件流	1. 学生进入“我的报修”列表。 2. 学生查看所有报修单的当前状态（如：待接单、维修中、已完成）。 3. [扩展点: 评价服务] 当工单状态为“已完成”时，学生可以选择 UC11 评价维修服务。
后置条件	学生获知了报修单的最新状态。

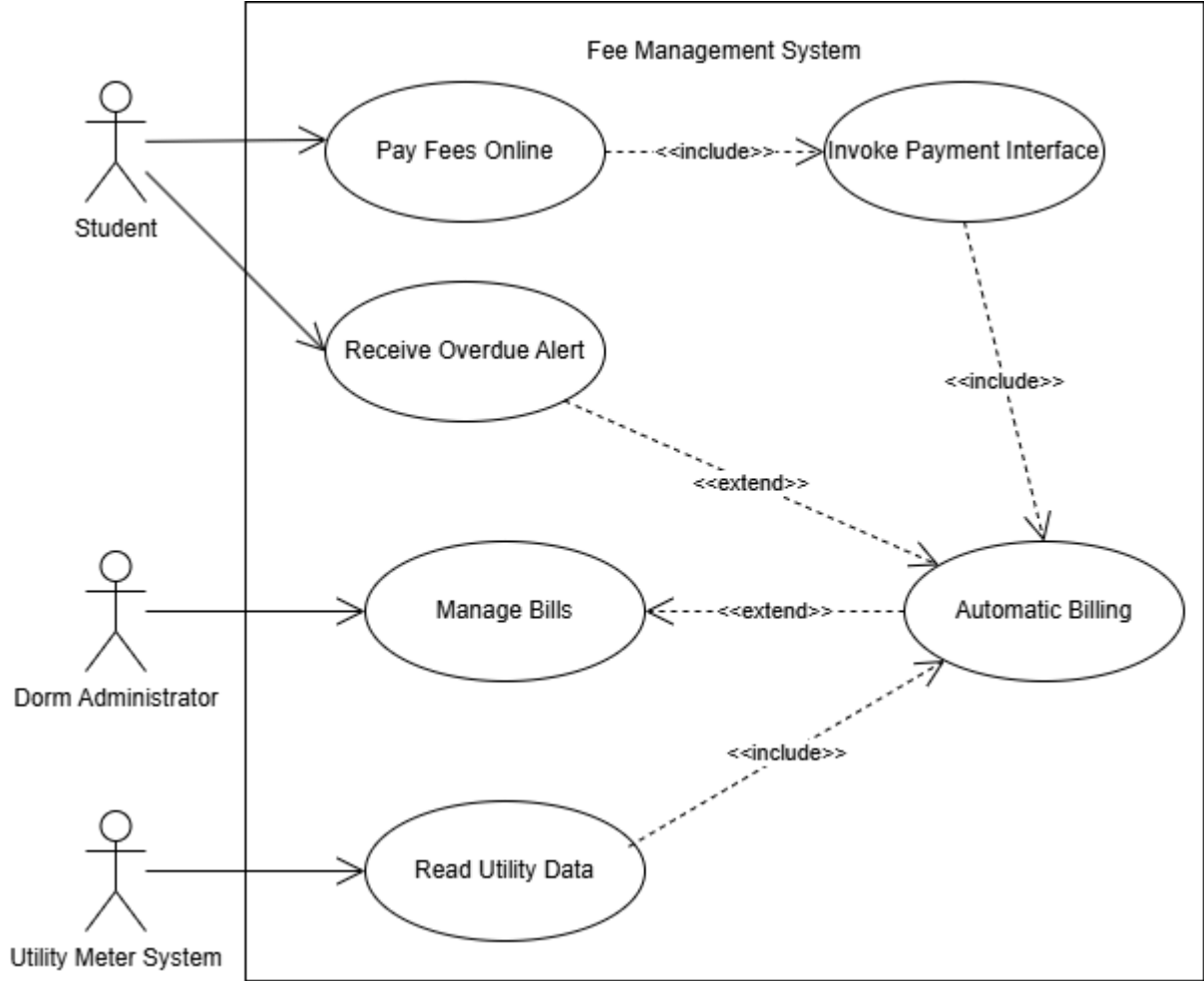
用例名: 评价维修服务 (UC11)

属性	描述
用例名	评价维修服务
编号	UC11
参与者	学生
前置条件	1. UC10 追踪维修进度 的工单状态为“已完成”。 2. 学生在 UC10 的扩展点选择“评价”。
事件流	1. [$\langle \rangle$] 本用例扩展 UC10 追踪维修进度。 2. 学生在 UC10 界面点击“评价”按钮。 3. 系统显示评价页面（如星级、评论框）。 4. 学生填写评价内容并提交。 5. 系统保存评价，并将其与工单和维修人员关联。
后置条件	1. 维修工单被标记为“已评价”。 2. 评价内容被保存，可用于 UC12 的监控。

用例名: 监控维修闭环 (UC12)

属性	描述
用例名	监控维修闭环
编号	UC12
参与者	宿管
前置条件	1. 宿管已登录系统。
事件流	1. 宿管进入“维修管理”或“数据看板”页面。 2. 宿管查看所有工单的概览（待处理、进行中等）。 3. [<>] 宿管使用 UC10 追踪维修进度 的功能来查看任一工单的详细状态和时间线。 4. 宿管查看维修人员效率和 UC11 的学生评价数据。
后置条件	宿管了解了维修工作的整体情况和效率。

缴费系统



用例名: 自动计费 (UC13)

属性	描述
用例名	自动计费
编号	UC13
参与者	时间 (系统定时触发)
前置条件	1. 系统已配置好计费规则（单价、周期）。 2. 水电表系统 接口可用。

属性	描述
事件流	<ol style="list-style-type: none"> 1. 时间 达到预设的计费点（如每月1日）。 2. 系统启动自动计费任务。 3. 系统遍历所有需计费的宿舍。 4. [$\langle \rangle$] 系统为每个宿舍执行 UC14 读取水电数据。 5. 系统根据获取的用量和单价，计算本周期费用，生成待缴账单。 6. 系统向相关学生推送新账单通知。
后置条件	<ol style="list-style-type: none"> 1. 所有宿舍的本周期账单已生成。 2. 学生可查询到新账单。
扩展场景	<ol style="list-style-type: none"> 4a. UC14 执行失败（如仪表离线）。 4a1. 系统标记该宿舍账单异常，并通知 宿管 (UC16A) 人工处理。

用例名: 读取水电数据 (UC14)

属性	描述
用例名	读取水电数据
编号	UC14
参与者	系统 (由 UC13 自动触发), 水电表系统
前置条件	<ol style="list-style-type: none"> 1. UC13 自动计费 被触发。 2. 水电表系统 接口可用。
事件流	<ol style="list-style-type: none"> 1. 系统接收到 UC13 的数据读取请求。 2. 系统调用 水电表系统 的接口，请求指定宿舍的当前表显读数。 3. 水电表系统 返回实时读数。 4. 系统将读数（及计算后的用量）返回给 UC13。
后置条件	<ol style="list-style-type: none"> 1. UC13 成功获取了水电用量数据。

用例名: 在线支付 (UC15)

属性	描述
用例名	在线支付
编号	UC15
参与者	学生
前置条件	<ol style="list-style-type: none"> 1. 学生已登录系统 (UC01)。 2. 学生至少有一笔待缴账单 (由 UC13 生成)。
事件流	<ol style="list-style-type: none"> 1. 学生进入“我的账单” (UC16A) 界面，选择一笔“待缴费”账单。 2. 学生确认金额，点击“立即支付”。 3. 学生选择支付方式（如微信、支付宝）。 4. [$\langle \rangle$] 系统执行 UC15A 调用支付接口。 5. UC15A 返回支付凭证（如二维码或App唤起）。 6. 学生在外部支付App中完成支付。 7. 系统接收到 支付渠道 的异步回调通知。 8. 系统验证通知，将账单状态更新为“已支付”。
后置条件	<ol style="list-style-type: none"> 1. 目标账单状态更新为“已支付”。 2. 学生收到支付成功通知。
扩展场景	<ol style="list-style-type: none"> 7a. 支付失败或超时。 7a1. 系统保持账单为“待缴费”状态，并提示学生“支付失败”。

用例名: 调用支付接口 (UC15A)

属性	描述
用例名	调用支付接口
编号	UC15A
参与者	系统 (由 UC15 自动触发), 支付渠道
前置条件	1. UC15 在线支付 被触发。 2. 支付渠道 接口已配置且可用。
事件流	1. 系统接收到 UC15 的支付请求 (包含金额、订单号)。 2. 系统调用 支付渠道 的统一下单接口。 3. 支付渠道 返回支付凭证 (如支付URL或App唤起参数)。 4. 系统将凭证返回给 UC15 的前端界面。
后置条件	1. UC15 获得了用于唤起支付的凭证。

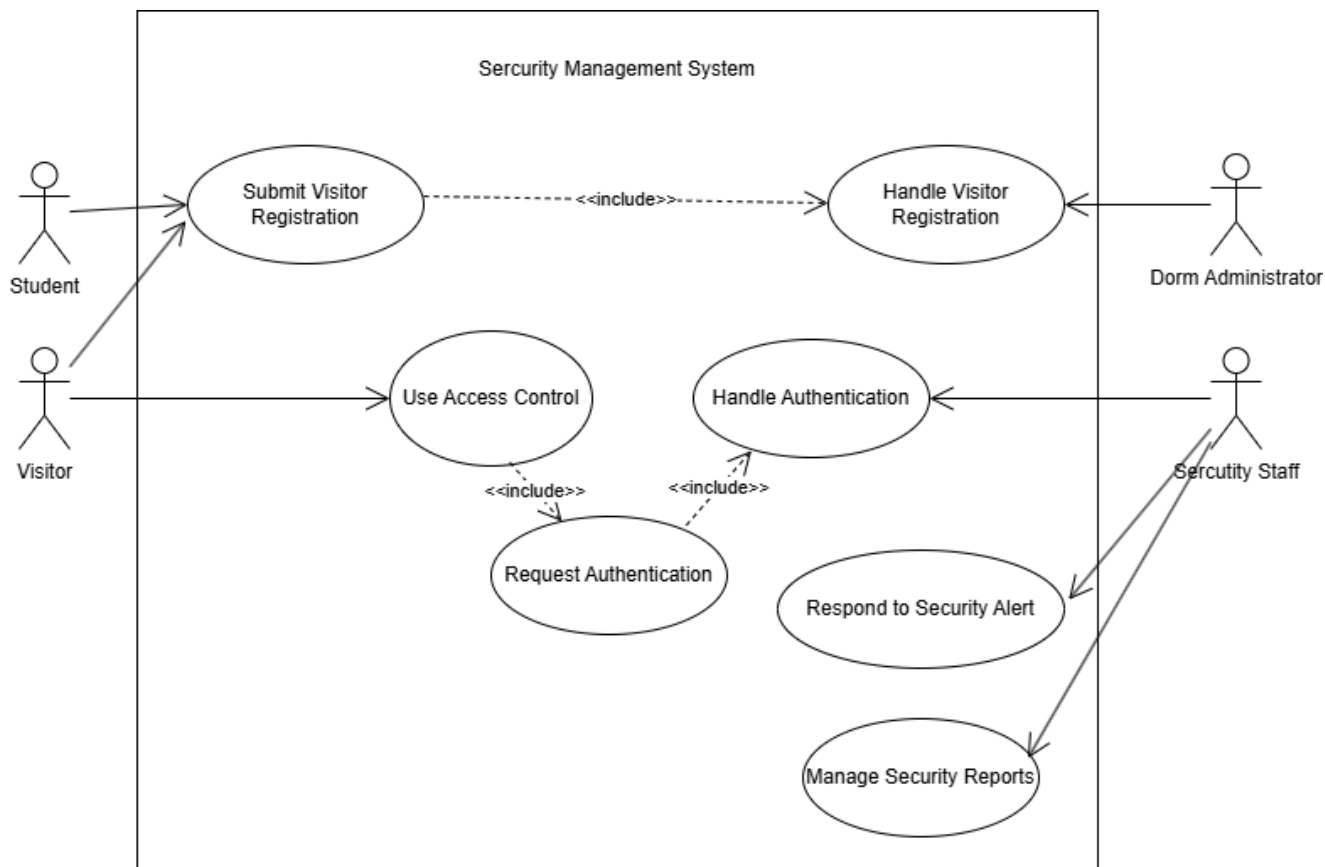
用例名: 管理账单 (UC16A)

属性	描述
用例名	管理账单
编号	UC16A
参与者	学生, 宿管
前置条件	1. 参与者已登录系统 (UC01)。
事件流	<p>[场景A: 学生查看账单]</p> <ol style="list-style-type: none"> 学生 进入“我的账单”页面。 系统显示该 学生 宿舍的所有历史账单 (包含状态: 待缴费、已支付、异常)。 学生 可按月筛选, 并查看账单详情 (如水电用量明细)。 <p>[场景B: 宿管管理账单]</p> <ol style="list-style-type: none"> 宿管 进入“费用管理”模块。 宿管 可按楼栋、按状态 (如“全部欠费”) 筛选和查询所有账单。 宿管 可手动修正 UC13 标记的异常账单。 宿管 可执行“一键催缴”操作 (触发 UC16B)。
后置条件	1. 学生 获知了自己的账单详情。 2. 宿管 掌握了整体缴费情况或处理了异常账单。

用例名: 接收欠费预警 (UC16B)

属性	描述
用例名	接收欠费预警
编号	UC16B
参与者	学生
前置条件	1. 学生存在“待缴费”账单且已过最后缴费期限。
事件流	<ol style="list-style-type: none"> 系统定时任务 (时间 参与者) 检测到学生账单已逾期。 (或 宿管 在 UC16A 中手动触发催缴)。 系统向该 学生 发送欠费预警通知 (如App推送、短信)。 学生 接收到通知。
后置条件	1. 学生 被提醒需要尽快缴费。

访客系统



用例名: 提交访客预约 (UC17A)

属性	描述
用例名	提交访客预约
编号	UC17A
参与者	学生, 访客
前置条件	1. 参与者 (学生) 已登录 (UC01)。 2. (或) 访客打开了公开的预约入口。
事件流	1. 参与者进入“访客预约”页面。 2. 参与者填写访客信息 (姓名、证件号、来访时间、事由、被访学生信息)。 3. 参与者确认并提交申请。 4. [<>] 系统自动包含 UC17B 处理访客登记 (的创建步骤), 生成一条“待审批”的访客记录。
后置条件	1. 系统中创建一条“待审批”的访客工单。 2. 系统通知 安保 或 宿管 进行审批。

用例名: 处理访客登记 (UC17B)

属性	描述
用例名	处理访客登记
编号	UC17B
参与者	安保, 宿管
前置条件	1. 参与者已登录系统 (UC01)。 2. 存在由 UC17A 提交的“待审批”访客记录。

属性	描述
事件流	<ol style="list-style-type: none"> 1. 安保 (或 宿管) 收到审批通知, 进入“访客管理”列表。 2. 参与者选择一条“待审批”记录查看详情。 3. [场景A: 批准] <ol style="list-style-type: none"> 3a1. 参与者点击“批准”。 3a2. 系统生成临时通行凭证 (如二维码) 并更新记录为“已批准”。 3a3. 系统向访客 (和学生) 发送批准通知及通行凭证。 4. [场景B: 驳回] <ol style="list-style-type: none"> 4a1. 参与者点击“驳回”并填写理由。 4a2. 系统更新记录为“已驳回”。 4a3. 系统向访客 (和学生) 发送驳回通知。
后置条件	<ol style="list-style-type: none"> 1. 访客记录状态被更新。 2. 访客和学生收到审批结果。

用例名: 使用门禁 (UC18A)

属性	描述
用例名	使用门禁
编号	UC18A
参与者	学生, 访客 (在 UC17B 批准后)
前置条件	<ol style="list-style-type: none"> 1. 参与者位于门禁读卡器或人脸识别设备前。 2. 参与者拥有有效的凭证 (校园卡、人脸或访客二维码)。
事件流	<ol style="list-style-type: none"> 1. 参与者将凭证靠近识别设备 (如刷卡、刷脸、扫码)。 2. 设备读取凭证信息。 3. [$\langle \rangle$] 系统自动包含并执行 UC18C 请求身份验证。 4. UC18C 返回“验证通过”。 5. 门禁解锁, 允许通行。
后置条件	<ol style="list-style-type: none"> 1. 参与者通过门禁。 2. 系统记录一条通行日志。
扩展场景	<ol style="list-style-type: none"> 4a. UC18C 返回“验证失败”。 4a1. 门禁保持锁定, 并提示“认证失败”。

用例名: (安保)处理认证 (UC18B)

属性	描述
用例名	(安保)处理认证
编号	UC18B
参与者	安保
前置条件	<ol style="list-style-type: none"> 1. 安保 已登录安防工作站 (UC01)。 2. 存在需要人工处理认证的情况 (如 UC18A 失败, 或访客忘带凭证)。
事件流	<ol style="list-style-type: none"> 1. 安保 在工作站上打开“人工认证”界面。 2. 安保 通过查询姓名或证件号, 调取访客/学生信息。 3. [$\langle \rangle$] 系统自动包含并执行 UC18C 请求身份验证 (查询该人员的当前状态)。 4. UC18C 返回该人员的状态 (如“有效”、“已批准访客”、“无效”)。 5. 安保 根据返回状态和人工核对 (如查看身份证件), 决定是否放行。 6. 安保 手动点击“开门”按钮。
后置条件	<ol style="list-style-type: none"> 1. 在 安保 的人工干预下, 门禁开启。 2. 系统记录一条人工认证日志。

用例名: 请求身份验证 (UC18C)

属性	描述
用例名	请求身份验证
编号	UC18C
参与者	系统 (由 UC18A 或 UC18B 自动触发), 门禁系统
前置条件	1. UC18A 或 UC18B 被触发。 2. 门禁系统 接口可用。
事件流	1. 系统接收到 UC18A 或 UC18B 传来的凭证信息或人员ID。 2. 系统查询内部数据库, 核对该人员的权限 (如: 是否住宿学生、是否在有效期内的访客)。 3. 系统将“允许”或“拒绝”的指令发送给 门禁系统 硬件。 4. 系统将验证结果 (“通过”/“失败”) 返回给调用方。
后置条件	1. 门禁硬件收到开/关指令。 2. 调用用例 (UC18A / UC18B) 获知了验证结果。

用例名: 响应异常预警 (UC19)

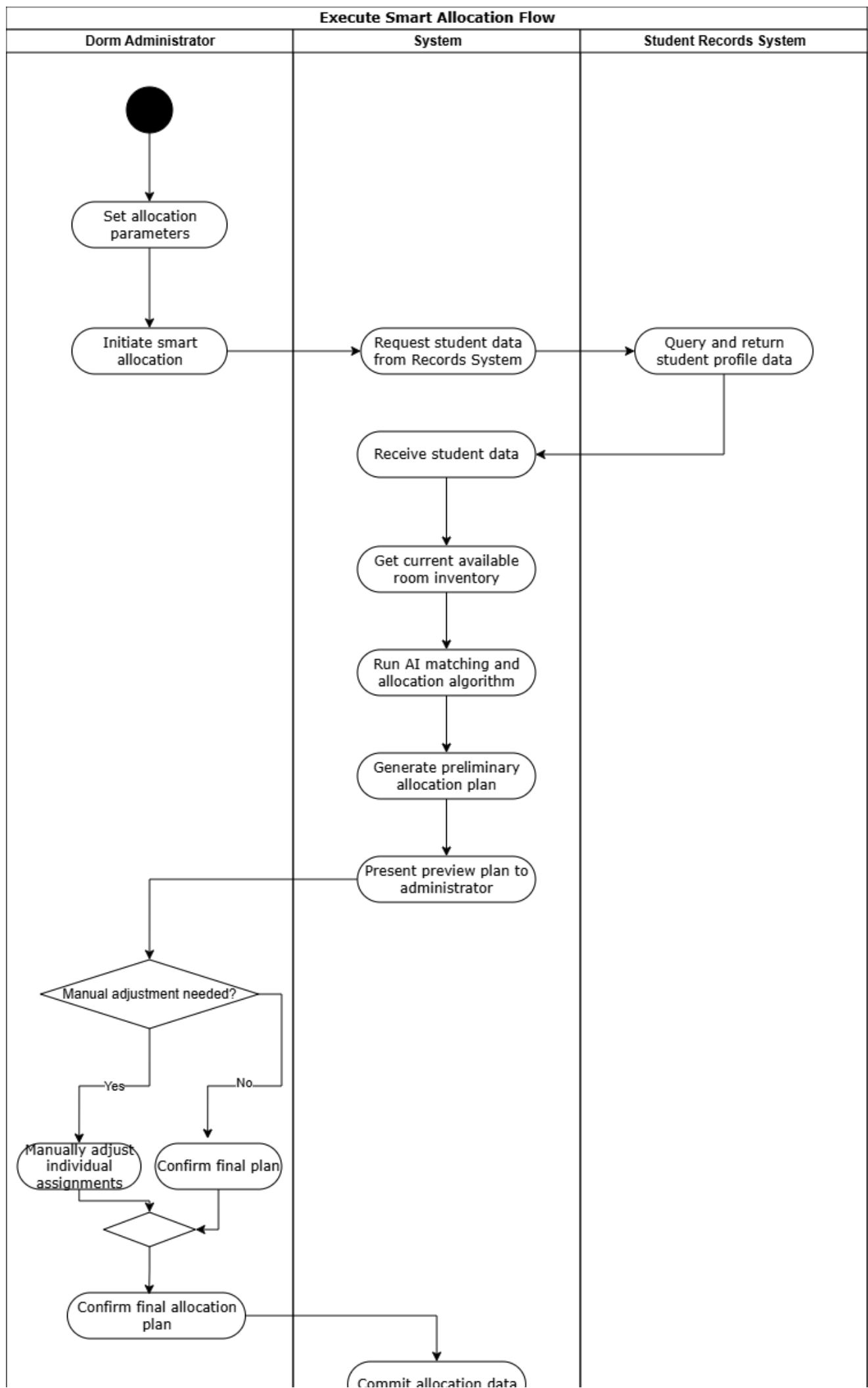
属性	描述
用例名	响应异常预警
编号	UC19
参与者	安保
前置条件	1. 安保 已登录安防工作站 (UC01)。 2. 系统 (如门禁或监控) 检测到异常行为。
事件流	1. 系统检测到异常 (如门禁被强行打开、多次认证失败、尾随闯入)。 2. 系统在 安保 工作站上弹出实时告警, 并 (可选) 联动摄像头。 3. 安保 收到告警, 立即查看告警详情和现场画面。 4. 安保 前往现场处理, 或通过广播系统发出警告。 5. 安保 在系统中处理该告警 (如: 确认为误报、填写处理结果)。
后置条件	1. 异常告警被处理。 2. 系统中留存一条告警处理记录。

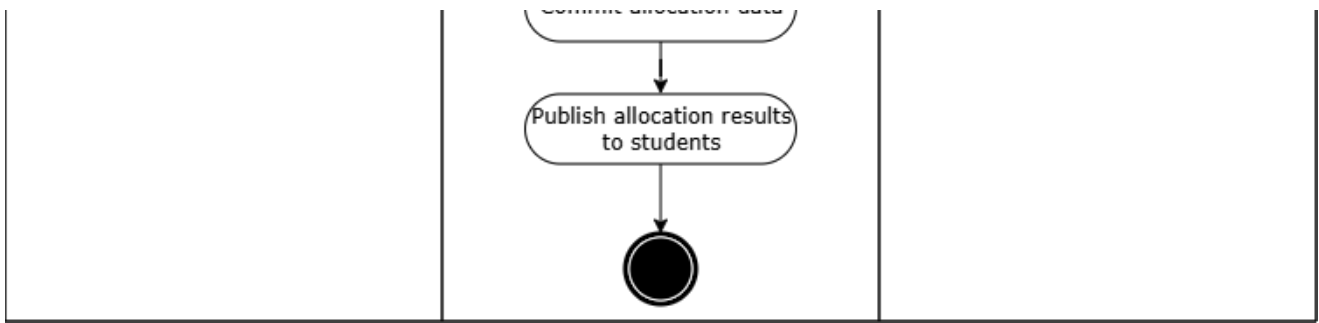
用例名: 管理安防报表 (UC20)

属性	描述
用例名	管理安防报表
编号	UC20
参与者	安保, 宿管
前置条件	1. 参与者已登录系统 (UC01)。
事件流	1. 安保 (或 宿管) 进入“数据报表”中心的“安防”模块。 2. 参与者选择报表类型 (如: 出入流量统计、异常告警汇总、访客记录)。 3. 参与者设置查询时间范围 (如: 过去24小时、上周)。 4. 系统生成并显示可视化图表及数据列表。 5. 参与者 (可选) 导出报表为Excel或PDF文件。
后置条件	1. 参与者获取了所需的安防统计数据。

4.3 活动图

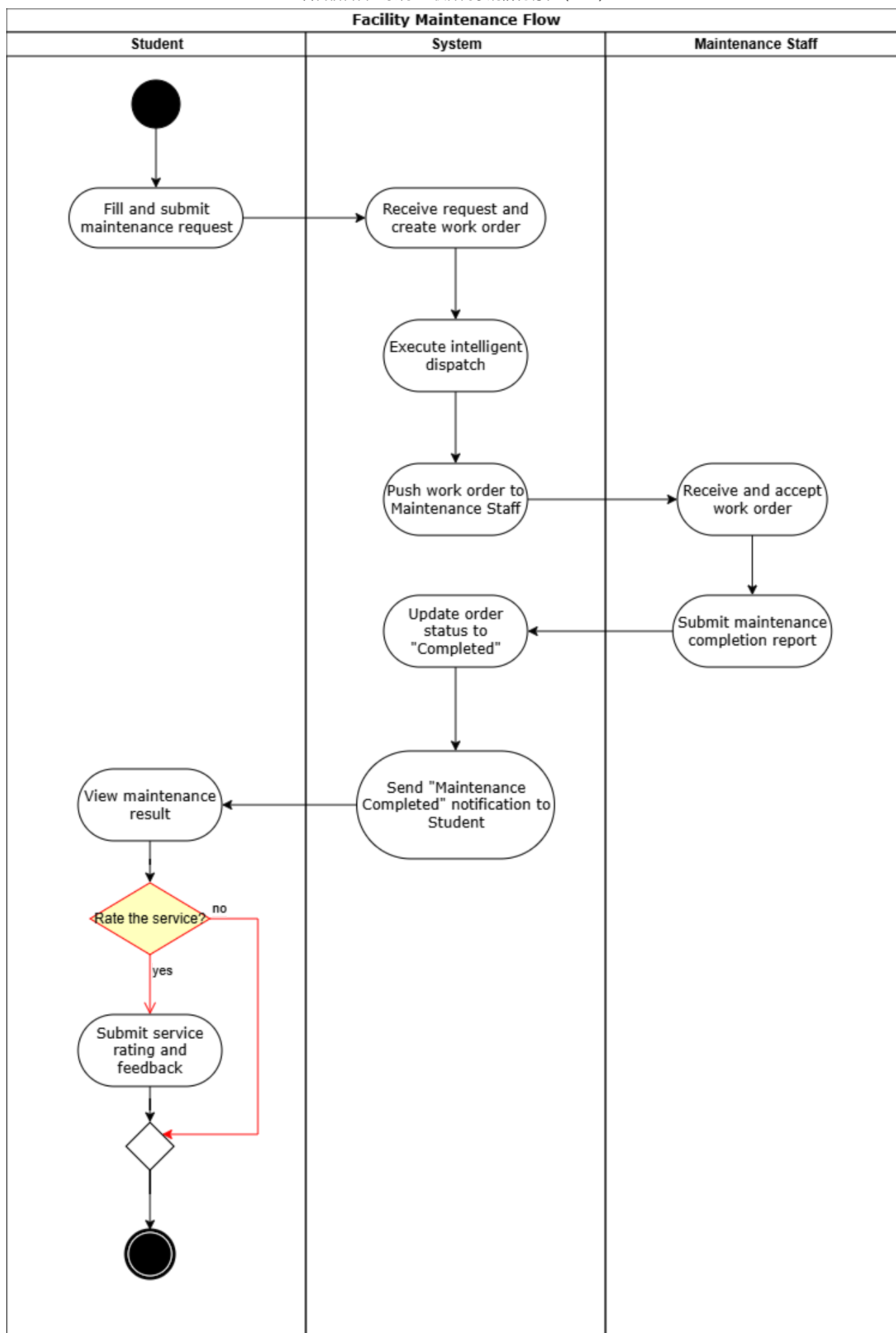
宿舍智能分配





“智能分配”流程由宿管 (Dorm Administrator) 启动。首先，宿管设置好分配参数并启动流程。系统 (System) 随即会向学籍系统 (Student Records System) 请求学生数据，在学籍系统返回数据后，系统会接收数据、获取当前可用房源，然后运行AI匹配与分配算法来生成初步方案。该方案将呈现给宿管审查。宿管此时判断是否需要人工调整：如果需要，宿管会手动调整分配结果；如果不需要，则直接确认。在宿管（调整后）“确认最终分配方案”后，系统会固化分配数据，并向学生发布最终的宿舍分配结果，整个流程随之结束。

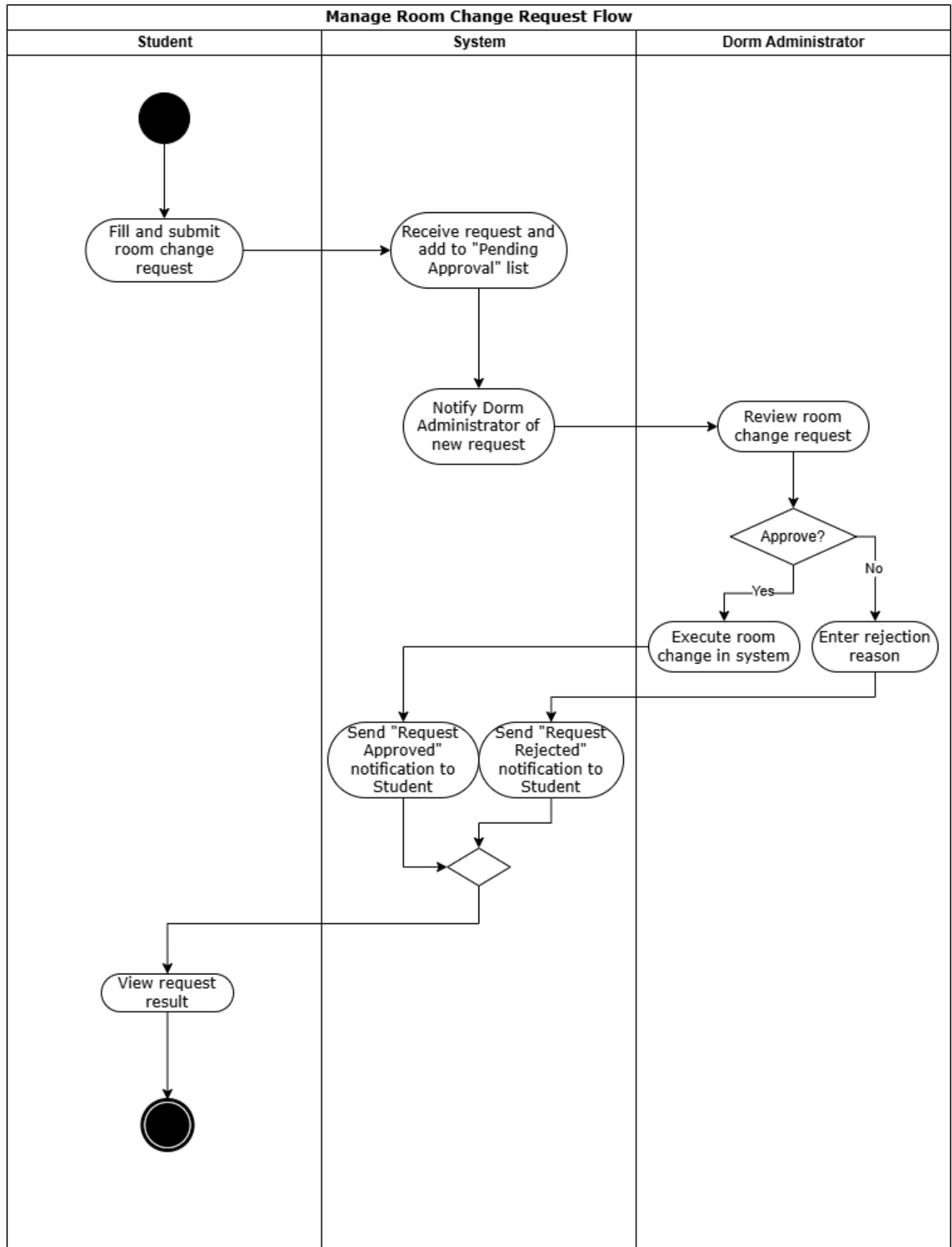
设备报修



整个流程由学生 (Student) 泳道发起。学生“填写并提交报修单”。系统 (System) 接收请求后，会“创建工单”并“执行智能派单”，随后“将工单推送给维修人员”。在维修人员 (Maintenance Staff) 泳道中，他们“接收并确认工单”。（在执行维修后）维修人员“提交维修完成报告”。流程回到系统泳道，系统将工单状态“更新为已完成”，并“向学生发送维修完成通知”。最后，流

程回到学生泳道。学生“查看维修结果”，并遇到一个“是否评价服务？”的决策点：如果选择“是”，学生将“提交服务评价和反馈”，然后流程结束；如果选择“否”，流程将直接结束。

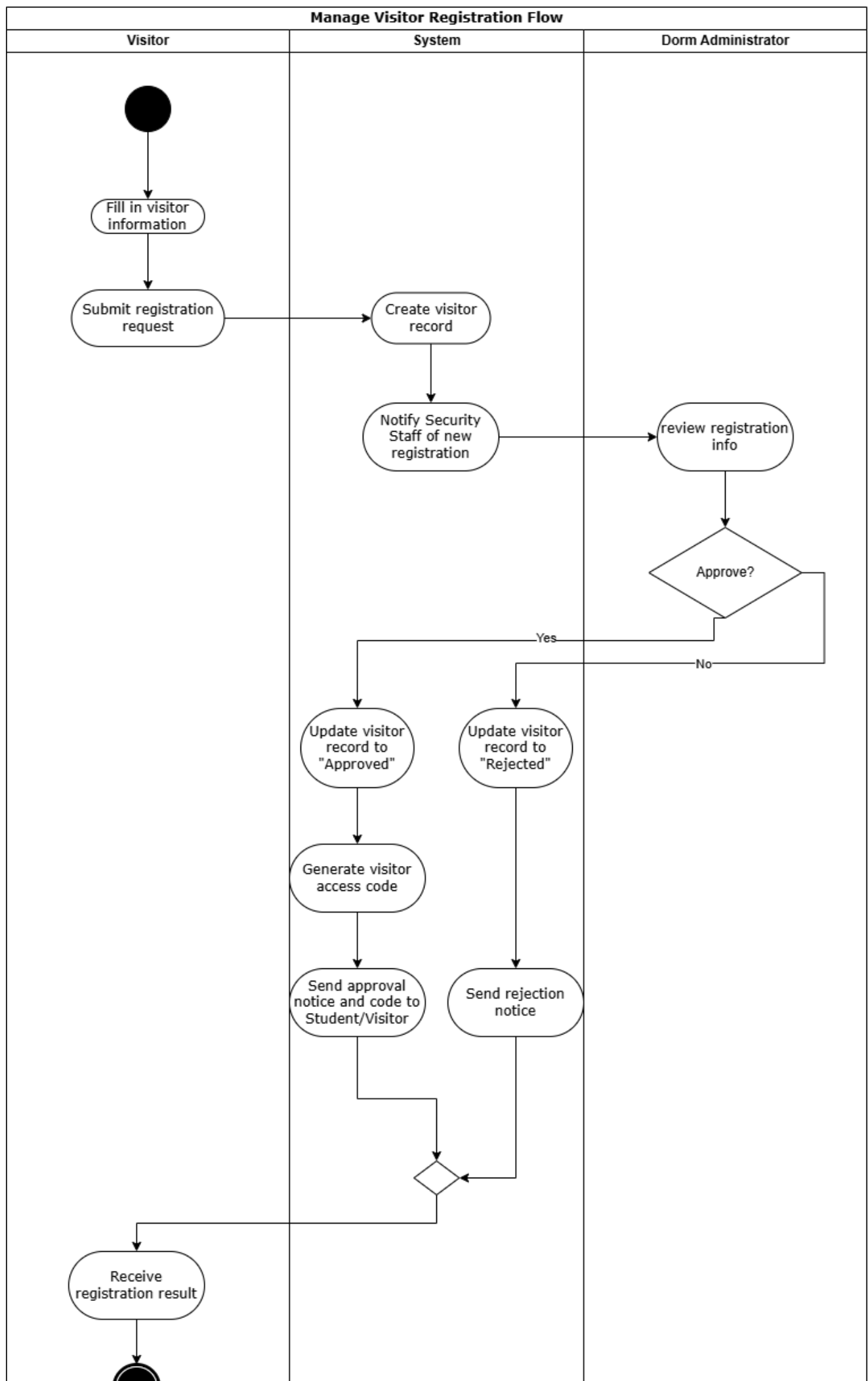
调宿申请管理



学生在系统里填写并提交一份调宿申请。系统收到后，会先把这个申请放进“待审批”列表，并马上通知宿管 (Dorm Administrator) 有新申请需要处理。宿管收到通知后，就会去审查这份申请的详细内容，然后做出决定。如果宿管点击“批准” (Yes)，他们就会在系统里操作调宿，然后系统会发一个“申请已批准”的通知给学生；但如果宿管选择“不批准”(No)，他们就需

要填写驳回的理由，系统会相应地发送“申请已驳回”的通知给学生。最后，无论申请是被批准还是被驳回，学生都会收到结果通知，并“查看最终的处理结果”，整个流程就结束了。

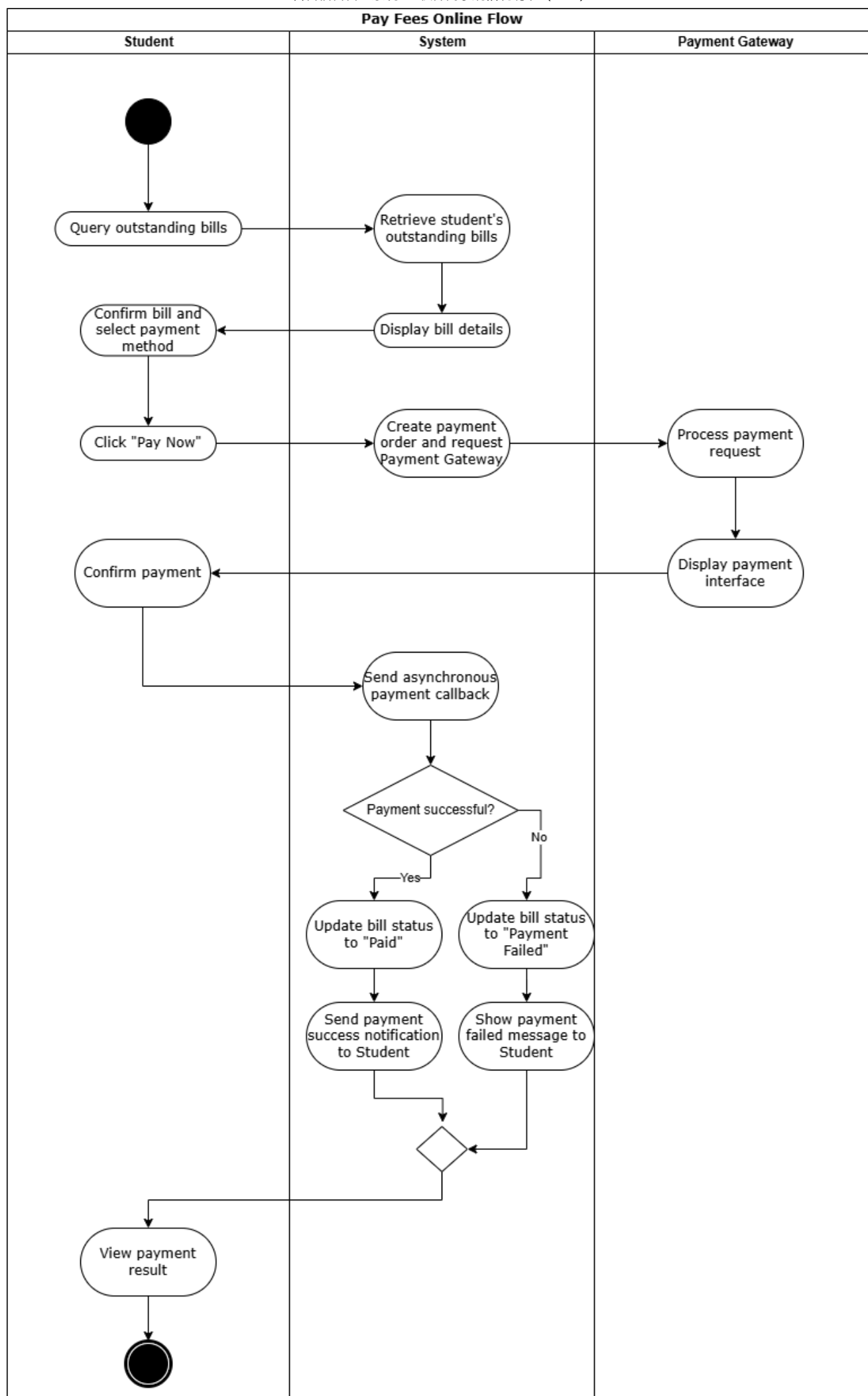
访客管理



		
--	--	--

访客 (Visitor) 在系统上“填写访客信息”并“提交预约请求”。系统 (System) 接收到请求后，会“创建一条访客记录”，并（根据流程）“通知宿管”(Dorm Administrator) 有新的登记需要处理。宿管收到通知后，就会“审查访客信息”，并决定是否“批准”(Approve?)。如果宿管批准了(Yes)，系统会自动把访客记录“更新为已批准”，并“生成一个访客通行码”，然后把这个“批准通知和通行码”发给访客或学生；如果宿管不批准(No)，系统则会把记录“更新为已驳回”，并只“发送一条驳回通知”。最后，访客都会“收到注册结果”，知道自己是否能来访，流程结束。

在线缴费



学生 (Student) 首先在系统里“查询自己的待缴账单”。系统 (System) 会马上“检索”这些账单并“显示详细信息”。学生核对无误后, “确认账单”并“选择支付方式”, 然后点击“立即支付”。这时, 系统会“创建订单”并转向支付网关 (Payment Gateway) “发起请求”。支付网关会“处理这个请求”并“弹出支付界面”让学生操作。学生在支付界面上“确认支付”后, 支付网关会“发送一个异步支付回调”通知系统。系统收到回调后会立刻判断“支付是否成功? ”: 如果成功了, 系统就把账单状态“更新为已支付”并给学生“发送成功通知”; 如果失败了, 系统就把账单“更新为支付失败”并“显示失败消息”给学生。无论哪种情况, 学生最后都会“查看到本次的支付结果”, 整个流程便完成了。

5 Glossary of Terms (术语表)

术语	英文对应	定义	相关用例
入住率	Occupancy Rate	实际入住床位数与总可用床位数的比例, 用于衡量宿舍资源利用效率	UC03, UC04
调宿	Room Transfer	学生从当前宿舍调整到其他宿舍的流程, 需经管理员审批	UC06A, UC06B, UC06C
派单	Work Order Dispatch	将维修任务分配给特定维修人员的过程, 支持智能自动派单	UC08, UC09A
SLA	Service Level Agreement	服务级别协议, 如报修处理承诺在24小时内完成的质量标准	UC07, UC12
能耗异常	Energy Consumption Anomaly	宿舍水电用量显著偏离正常模式的情况, 触发预警机制	UC13, UC14
门禁日志	Access Control Log	记录所有人员进出宿舍楼的时间、方式和身份信息	UC18A, UC18C, UC20
访客授权	Visitor Authorization	访客经过审批后获得的临时进出权限, 包括有效期限制	UC17A, UC17B
欠费提醒	Overdue Payment Reminder	针对逾期未缴费用的自动通知机制, 支持多渠道发送	UC16A, UC16B
限界上下文	Bounded Context	系统架构中明确边界的功能领域, 如分配、维修、缴费等	全系统
角色权限	Role-Based Access Control	基于用户角色分配系统操作权限的安全控制机制	UC01, UC02
智能分配	Smart Allocation	基于算法规则自动匹配学生与宿舍的优化分配过程	UC03, UC05
维修闭环	Maintenance Closed Loop	从报修提交到完成评价的全流程跟踪管理	UC07-UC12

6 Supplementary Specification (补充规格)

本章节定义了系统的非功能性需求, 这些需求是确保系统质量、性能和用户体验的关键。

类别 (CATEGORY)	指标 (INDICATOR)	验收标准 (ACCEPTANCE CRITERIA)
性能 (Performance)	响应时间	- 核心页面 (如登录、主页、信息查看) 加载时间 < 2秒。 - 关键操作 (如提交报修、查询数据) 后台处理响应时间 < 500毫秒。
	并发用户数	- 系统在高峰时段 (如选房、缴费) 能稳定支持至少 500 个并发用户同时在线操作, 无明显延迟或服务降级。
易用性 (Usability)	首次使用成功率	- 新用户在外无外部Helpers, 首次尝试完成核心任务 (如提交报修、查找宿舍信息) 的成功率 > 90%。
	任务完成效率	- 用户完成高频任务 (如缴费、查看通知) 的平均点击次数应比传统流程减少 50% 以上。
可靠性 (Reliability)	系统可用性	- 核心服务年度可用性 > 99.9%, 即全年计划外停机时间 < 8.76 小时。

类别 (CATEGORY)	指标 (INDICATOR)	验收标准 (ACCEPTANCE CRITERIA)
	容错性	- 单个非核心微服务（如通知服务）故障不影响核心业务（如登录、报修）的正常运行。
安全性 (Security)	数据安全	- 所有用户敏感数据（如密码、身份证号）在数据库中必须加密存储。 - 前后端数据传输必须使用 HTTPS/TLS 加密。
	访问控制	- 严格遵循基于角色的访问控制（RBAC），任何用户都不能越权访问未授权的接口或数据。
可维护性 (Maintainability)	代码质量	- 后端 Go 代码单元测试覆盖率 > 70%。 - 前端 TypeScript 代码无编译错误，并通过 ESLint 规范检查。
	部署效率	- 基于 Docker 的新版本部署或回滚操作应在 15 分钟内完成。
可扩展性 (Scalability)	架构扩展	- 系统应支持水平扩展，当用户量增加时，可通过增加服务实例来提升处理能力。 - 新增一种角色（如后勤主管）及其权限配置的开发周期 < 3 人日。

7 UI Mock-ups (界面快照)

本章节展示了系统主要界面的原型快照，旨在说明宿舍管理系统的核心交互逻辑与用户体验设计。界面均基于实际业务流程绘制，体现宿舍信息管理、报修、公告发布等主要功能。所有界面文件存放于 assignment1/src/UI/ 目录下，文件命名遵循格式 {PageName}_v{Version}.png。

7.1 登录页面 (Login Page)



说明： 用户通过输入学号与密码登录系统，可选择找回密码或查看使用指南。页面突出简洁性与易用性，确保在不同设备上均可良好显示。

7.2 个人宿舍主页 (Student Dorm Dashboard)



说明： 展示学生的宿舍基本信息、公告通知与报修记录。支持实时查看宿舍状态与报修进度，页面结构清晰，信息层次分明。

7.3 报修申请页面 (Repairment Request Page)

智舍系统宿舍物品报修主页



报修申请 报修记录 个人中心



报修申请表

报修类型:

下拉选择

问题描述:

上传照片:

选择文件

未选择任何文件

联系电话:

提交申请

重置

说明： 学生可选择报修类型、填写问题描述并上传相关照片。界面包含表单验证、上传状态与提交反馈提示，支持多端响应式布局。

7.4 宿舍管理员主界面 (Manager Dashboard)

宿舍管理员主界面



情况统计



报修管理



宿舍分配

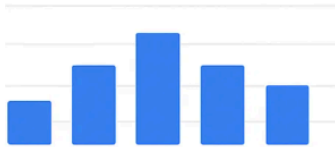


公告

管理员



报修统计



报修任务列表

编号	宿舍	内容	状态
#101	5-302	水管破裂	已完成
#102	8-204	门锁破坏	进行中

公告管理列表

新建公告

编辑公告

说明：宿舍管理员可查看报修任务列表与统计图表，并进行公告管理。界面设计突出信息汇总与操作高效性，提供“新增公告”“编辑公告”等关键交互按钮。

7.5 易用性与无障碍性说明

系统界面设计遵循高校信息化建设的无障碍规范，重点体现在以下方面：

- 视觉对比度优化：**深色标题栏与浅色背景的配色确保文字可读性；
- 交互一致性：**导航栏与表单控件布局统一，减少用户学习成本；
- 响应式布局：**所有页面均适配不同分辨率设备，支持桌面端与平板端访问；
- 辅助信息提示：**表单错误、加载中、空态页面均有明确提示信息。

7.6 章节总结

本系统的 UI 原型以简洁明快、结构清晰为设计核心，配合合理的交互流程和无障碍性考虑，确保了不同角色用户（学生、宿管、维修人员等）都能快速上手并完成日常操作。后续阶段将基于这些界面进行前端开发与用户测试，以持续优化交互体验。

8 AI Tools Usage Acknowledgement (ai工具使用声明)

在本项目文档撰写过程中，我们使用了AI工具辅助完成部分工作：

使用工具：DeepSeek Chat (2025年1月版本)

使用范围与用途：

- **头脑风暴：**辅助完善思路，帮助优化战略分析部分
- **格式规范化：**协助检查参考文献格式和术语定义的规范性

使用说明：

- AI生成的内容均经过团队成员审核和修改，确保符合项目实际情况
- 术语定义和战略分析基于项目真实需求和团队讨论结果
- 所有关键决策和业务目标均由团队成员独立完成

9 References (参考文献)

1. [1]Dennis, A., Wixom, B. H., & Tegarden, D. (2015). *Systems Analysis and Design: An Object-Oriented Approach with UML* (6th ed.). Wiley.

- 这本书介绍了使用面向对象方法进行系统分析和设计的过程，以及针对复杂系统开发的详细UML建模技术，提供了理解用例图、活动图和类模型的有价值框架。

[2]Sommerville, I. (2011). *Software Engineering* (9th ed.). Pearson Education.

- 提供关于软件工程实践的全面指南，包括系统可靠性、性能指标和设计约束。

[3]Pressman, R. S., & Maxim, B. (2014). *Software Engineering: A Practitioner's Approach* (8th ed.). McGraw-Hill Education.

- 讨论了软件开发模型、安全要求和大规模系统的可维护性。

[4]IBM. (1999). *IBM Common User Access (CUA) Guidelines*.

- 制定了用户界面设计的标准，以促进易用性、一致性和容错性。

[5]Myers, G. J., Sandler, C., & Badgett, T. (2011). *The Art of Software Testing* (3rd ed.). John Wiley & Sons.

- 重点介绍软件测试技术，包括保持低错误率和确保系统可靠性的方法。

[6]Bass, L., Clements, P., & Kazman, R. (2012). *Software Architecture in Practice* (3rd ed.). Addison-Wesley.

- 解释了创建可维护和可扩展软件系统的架构原则，重点关注性能和安全性。

[7]Jansen, A., & Bosch, J. (2005). *Software Architecture as a Set of Architectural Design Decisions*. Proceedings of the 5th Working IEEE/IFIP Conference on Software Architecture (WICSA'05).

- 讨论了影响系统可维护性和可扩展性的设计决策。

[8]ISO/IEC. (2001). *ISO/IEC 9126-1: Software Engineering – Product Quality – Part 1: Quality Model*. International Organization for Standardization.

- 提供评估软件质量属性的国际标准，包括可用性、性能和可靠性。

[9]LaToza, T. D., & Myers, B. A. (2010). *Developers Ask Reachability Questions*. ICSE '10: Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering, Volume 1.

- 解决了软件维护中的常见挑战，并介绍了改善系统容错性和设计约束的策略。

10 Contributions of Team Members (成员贡献)

本文档由智舍团队成员协作完成，各成员的具体分工与贡献如下：

姓名	学号	负责章节	主要贡献
张峻搏	2252964	3, 6, 10, 全文	组长。 负责制定项目路线图 (Roadmap)，构建非功能性需求框架，并整合团队成员贡献。同时承担全文的统稿、审校与一致性检查工作。
黄毅成	2252634	4	用例与流程负责人。 负责系统的核心用例建模，绘制了整体用例图，并为关键业务流程（如用户注册、宿舍分配、设备报修等）进行了详细描述和活动图/BPMN建模。
杨光	2251756	2, 5, 8, 9	战略与术语负责人。 负责项目的战略分析 (SWOT/TOWS)，定义了项目术语表，整理了参考文献，并撰写了AI工具使用声明。
马敏慧智	2351707	1, 7	UI与用户价值负责人。 负责撰写项目简介初稿，设计并提供了核心功能的用户界面 (UI) 快照，并为易用性与无障碍相关的非功能性需求进行了定义。