



北京交通大学
BEIJING JIAOTONG UNIVERSITY

航空乘客信息管理系统 系统规划报告

学期：2024-2025 第二学期

编制日期：2025 年 05 月 10 日

编制人：江家玮

学号：22281188

班级：计科 2204

目录

1. 引言	1
1.1 项目概述与目标	1
1.2 报告目的与范围	1
1.3 定义与缩写	1
2. 系统规划详述	2
2.1 系统名称	2
2.2 业主企业组织架构与系统关联部门	2
2.2.1 业主企业/组织机构概述	2
2.2.2 组织架构图	2
2.2.3 系统涉及的相关业务部门	3
2.3 用户与业务场景分析	4
2.3.1 系统主要用户角色分析	4
2.3.2 用户使用系统开展业务的场景描述	5
2.4 系统性能指标规划	8
2.5 系统战略地位、投资成本与预期收益	9
2.5.1 系统战略地位	9
2.5.2 建设投资与运营成本估算	11
2.5.3 预期收益分析	12
2.6 初步技术选型规划与技术可行性分析	13
2.6.1 初步技术选型规划	13
2.6.2 技术可行性分析	14
3. 函数依赖、范式及 3NF 分解	15
3.1 函数依赖分析	15
3.2 范式与规范化	16
3.2.1 第一范式 (1NF)	16
3.2.2 第二范式 (2NF)	16
3.2.3 第三范式 (3NF)	16
3.2.4 BCNF	17
3.3 从不符合 3NF 的模式分解	17
3.4 变换后的 ER 图	18

1. 引言

1.1 项目概述与目标

本项目旨在规划与设计一个现代化、高效的“航空乘客信息管理系统”(Airline Passenger Information Management System, APIMS)。在全球航空业日益依赖信息化手段提升竞争力的背景下，一个能够集中管理乘客数据、优化票务流程、支持客户服务的系统显得至关重要。APIMS 通过集成化的数据库应用，实现乘客信息的快速准确录入、便捷查询、灵活修改与安全删除，并为未来的业务扩展(如在线预订、客户关系管理、数据分析决策)奠定坚实的技术基础。本规划报告将详细阐述系统的各项规划要素，包括系统命名、组织架构与业务关联、用户场景分析、性能指标设定、战略定位与成本效益分析，以及初步的技术选型与可行性评估。

1.2 报告目的与范围

本系统规划报告的主要目的是：

- ✓ 明确 APIMS 的建设目标、核心功能和战略价值。
- ✓ 分析系统所服务的组织机构、涉及的业务部门及各类用户的使用场景。
- ✓ 规划系统的关键性能指标，确保系统能够满足实际运营需求。
- ✓ 评估系统建设的潜在投资、运营成本及预期收益，进行初步的经济可行性分析。
- ✓ 为项目下一阶段的详细需求分析、系统设计和开发实施提供清晰的指导蓝图。

本报告的范围涵盖了 APIMS 的系统级规划，包括业务背景分析、功能与性能规划、以及初步的技术和经济可行性考量。

1.3 定义与缩写

- ✓ APIMS: Airline Passenger Information Management System (航空乘客信息管理系统)
- ✓ CRUD: Create, Read, Update, Delete (创建、读取、更新、删除) - 数据库基本操作。

- ✓ RDBMS: Relational Database Management System (关系型数据库管理系统)。
- ✓ UI: User Interface (用户界面)。
- ✓ UX: User Experience (用户体验)。
- ✓ API: Application Programming Interface (应用程序编程接口)。
- ✓ IT: Information Technology (信息技术)。
- ✓ CRM: Customer Relationship Management (客户关系管理)。
- ✓ GDS: Global Distribution System (全球分销系统)。
- ✓ OTA: Online Travel Agency (在线旅行社)。

2. 系统规划详述

2.1 系统名称

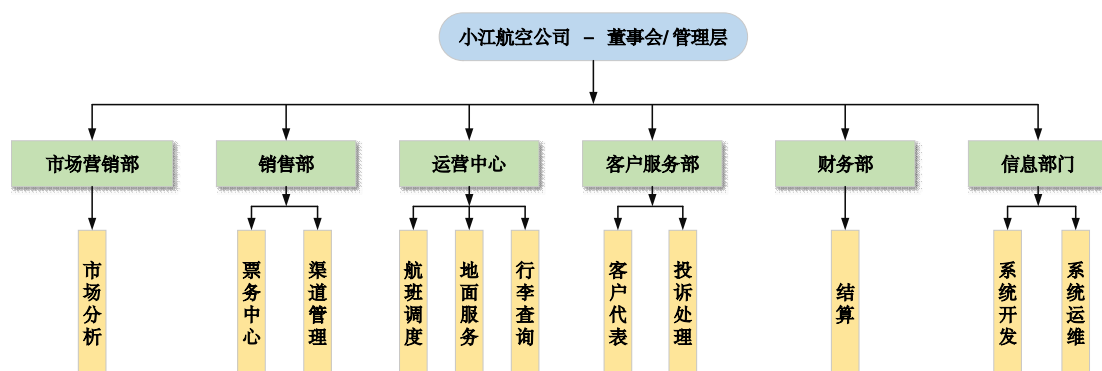
经过综合考量系统的核心功能与业务领域，本数据库应用系统正式命名为：航空乘客信息管理系统 (Airline Passenger Information Management System)，英文缩写为 APIMS。该名称直观地反映了系统的主要服务对象（航空公司）、核心管理内容（乘客信息）以及其系统属性。

2.2 业主企业组织架构与系统关联部门

2.2.1 业主企业/组织机构概述

为便于规划，我假设 APIMS 的业主企业为一家中等规模的航空公司，例如“小江航空公司”。该公司拥有国内及部分国际航线，致力于通过提升信息化水平来优化运营和客户服务。

2.2.2 组织架构图



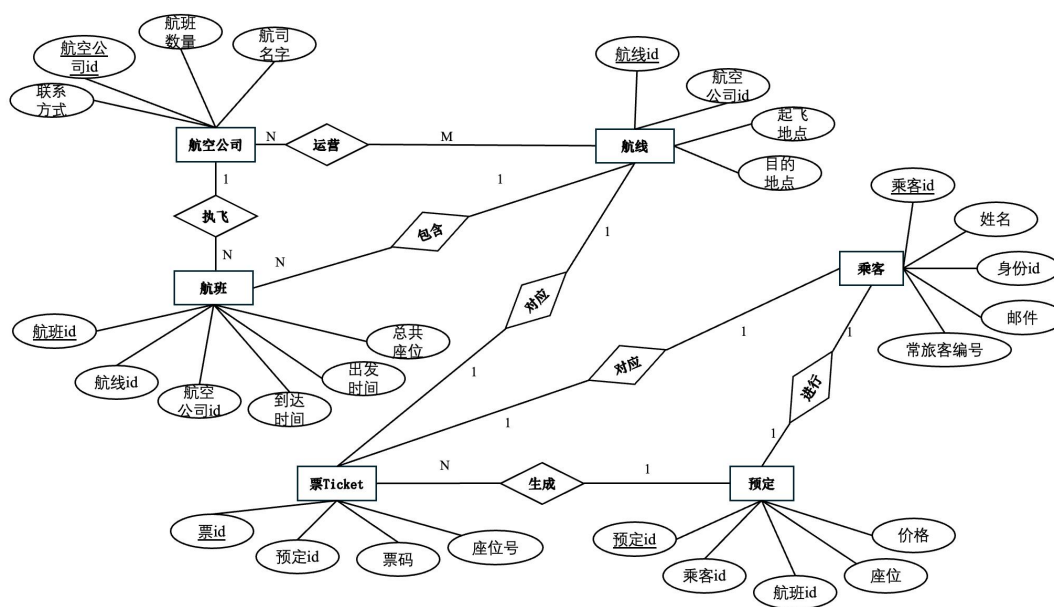
2.2.3 系统涉及的相关业务部门

APIMS 的建设和运营将紧密关联航空公司内的多个核心业务部门，主要包括：

- 销售部 (Sales Department):
 - ✓ **票务代理：** 这是 APIMS 最直接和最主要的用户群体。他们将使用系统进行乘客信息的录入、查询（例如，核实乘客身份、查询历史预订）、修改（例如，更新乘客联系方式）、以及可能的机票预订、改签和退票操作。APIMS 是他们日常工作的核心工具。
 - ✓ **渠道管理：** 若系统未来与 GDS、OTA 等外部销售渠道对接，渠道管理部门也需要关注系统的数据接口和销售数据统计。
- 客户服务部 (Customer Service Department):
 - ✓ **客服代表：** 在处理乘客咨询、投诉或特殊服务请求时，客服代表需要通过 APIMS 快速准确地查询乘客信息、历史出行记录、常旅客状态等，以便提供个性化和高效的服务。
 - ✓ **常旅客计划管理：** APIMS 存储的乘客信息（尤其是常旅客编号和出行频次）是常旅客计划管理的基础。
 - ✓ **投诉处理/失物招领：** 可能需要查询乘客信息以协助处理。
- 运营中心 (Operations Center):
 - ✓ **地面服务部门（值机、登机口）：** 虽然 Lab5 的核心系统未直接覆盖这些功能，但一个完整的乘客管理体系最终会延伸至此。地服人员需要 APIMS（或其集成系统）来核验乘客身份、票务状态、行李信息等。
 - ✓ **航班调度：** 航班信息是乘客出行的基础，APIMS 中的乘客信息与航班信息紧密关联。航班的任何变动（延误、取消）都需要及时反映并通知到相关乘客，这可能需要 APIMS 与航班调度系统的数据联动。
- 市场营销部 (Marketing Department):
 - ✓ APIMS 积累的乘客数据（如出行偏好、消费习惯、客源地分布等）是市场分析和精准营销的重要依据。市场部门可以通过系统（或其数据导出功能）获取数据支持，以制定更有效的营销策略和产品推广活动。
- 信息技术部 (IT Department):
 - ✓ 作为系统的建设和运维部门，IT 部将全程参与 APIMS 的需求分析、设计、开发、测试、部署、日常维护、安全管理和后续升级工作。数据库管理员 (DBA) 将负责数据库的性能优化和数据安全。

➤ 财务部 (Finance Department):

- ✓ 机票销售产生的收入与乘客信息和预订记录直接相关。财务部门可能需要从 APIMS (或其关联的销售/结算系统) 获取票务销售数据, 进行账目核对和收入结算。
- ✓ 通过 APIMS 的建设, 可以打破各部门间的信息壁垒, 实现乘客信息在授权范围内的共享和高效流转, 从而提升航空公司的整体运营协同效率。



2.3 用户与业务场景分析

2.3.1 系统主要用户角色分析

基于前述部门关联及前期的实验报告的用户角色设定, APIMS 的主要用户角色及其特征如下:

➤ 票务代理 (Ticket Agent):

- ✓ **特征**: 系统的一线操作员, 日常工作高度依赖此系统。需要快速、准确地完成乘客信息处理和票务操作。对系统的易用性和响应速度要求高。
- ✓ **核心需求**: 乘客信息的增删改查、机票预订与管理。

➤ 航班管理员 (Flight Manager):

- ✓ **特征**: 负责航班数据维护的专业人员。对数据的准确性和及时性要求高。
- ✓ **核心需求**: 航班时刻、状态、机型等信息的录入、修改与查询。其工作成果是票务代理进行预订的基础。

➤ 数据分析师 (Data Analyst):

- ✓ **特征:** 需要从系统中提取和分析数据, 为管理层提供决策支持。需要灵活的数据查询和导出功能。
- ✓ **核心需求:** 对乘客数据、销售数据、航班运营数据进行多维度查询、统计和分析。

➤ 系统管理员 (System Administrator):

- ✓ **特征:** IT 技术人员, 负责系统的稳定运行和安全。
- ✓ **核心需求:** 用户账户管理、权限控制、数据备份与恢复、系统监控与故障排除。

➤ 乘客 (Passenger):

- ✓ **特征:** 航空服务的最终消费者。如果未来系统扩展提供在线自助服务(如官网/APP 订票、在线值机、信息查询), 乘客将成为系统的外部用户。他们期望操作便捷、信息透明、个性化服务。
- ✓ **核心需求 (若有自助服务):** 查询航班、在线预订、管理个人行程、在线值机、查询常旅客积分等。

2.3.2 用户使用系统开展业务的场景描述

➤ 场景一: 票务代理为新乘客预订国内航班机票

- ✓ **触发:** 乘客李先生通过电话联系航空公司票务中心, 希望预订一张从北京到上海的单程机票, 日期为下周三。
- ✓ **票务代理操作:**
 - ◇ 票务代理王小姐接听电话, 打开办公电脑上的 APIMS 系统, 使用个人工号和密码登录。
 - ◇ 进入系统主界面后, 王小姐点击“航班查询”功能模块。
 - ◇ 在查询界面, 输入出发城市“北京”, 到达城市“上海”, 出发日期“2025-05-21”。
 - ◇ 系统显示符合条件的航班列表, 包括航班号、起降时间、票价、剩余座位数。
 - ◇ 王小姐向李先生报读可选航班信息。李先生选择了 CA1501 航班, 10:00 起飞。
 - ◇ 王小姐在系统中选中 CA1501 航班, 点击“预订”按钮。

- ◇ 系统提示输入乘客信息。王小姐询问李先生是否首次乘坐本公司航班。李先生表示是首次。
 - ◇ 王小姐点击“添加新乘客”选项。APIMS 弹出乘客信息录入表单。
 - ◇ 王小姐根据李先生口述，依次输入其姓名“李明”、身份证号“11010119900307XXXX”、手机号码“138XXXXXXXX”。邮箱地址李先生选择不提供。
 - ◇ 王小姐点击“保存乘客信息”。系统校验通过（如身份证号格式正确且未在系统中重复），提示乘客信息保存成功。
 - ◇ 返回预订界面，系统自动带入刚录入的乘客李明的信息。王小姐再次与李先生确认航班、日期、乘客信息无误。
 - ◇ 王小姐点击“确认预订”。系统检查该航班余票充足，生成预订记录，状态为“待支付”。系统同时生成一个预订编号 PNR-20250510-XXXX。
 - ◇ 王小姐告知李先生预订成功，预订编号及票价，并提醒其在规定时间内完成支付。同时，系统可能会自动发送一条包含预订信息的短信给李先生（若集成短信服务）。
- ✓ 结束：票务代理完成一次新乘客的机票预订信息录入。

➤ 场景二：航班管理员更新航班延误状态

- ✓ 触发：航空公司运营控制中心（AOC）通知，由于航路天气原因，原定今日 14:00 从广州飞往成都的 ZH9876 航班预计延误至 16:30 起飞。
- ✓ 航班管理员操作：
 - ◇ 航班管理员张工登录 APIMS 系统（或其集成的航班管理模块）。
 - ◇ 进入“航班状态管理”界面。
 - ◇ 通过航班号“ZH9876”和日期查询到该航班记录。
 - ◇ 选中该航班，点击“更新状态”按钮。
 - ◇ 在弹出的状态更新窗口中，将航班状态从“计划 (Scheduled)”修改为“延误 (Delayed)”。
 - ◇ 在“预计起飞时间”字段，将时间更新为当日 16:30。
 - ◇ 在“延误原因”字段，选择或输入“天气原因-航路雷雨”。
 - ◇ 点击“确认更新”。系统保存新的航班状态和信息。
- ✓ 后续影响（系统联动）：

- ◇ APIMS（或通过接口）将此航班延误信息推送给已预订该航班的乘客的联系方式（短信/APP 通知）。
- ◇ 票务代理在查询该航班时，将看到更新后的延误状态和预计起飞时间。

✓ 结束： 航班管理员完成航班延误状态的更新。

➤ 场景三：数据分析师查询季度乘客客源地分布

- ✓ 触发： 市场部经理需要了解上一季度公司主要航线的乘客客源地分布情况，以调整区域营销策略。
- ✓ 数据分析师操作：
 - ◇ 数据分析师刘女士登录 APIMS 系统（或其数据分析模块/报表工具）。
 - ◇ 选择“乘客数据分析”或“自定义报表”功能。
 - ◇ 设置查询参数：时间范围（例如：2025 年第一季度），航线范围（例如：选择北京-上海、广州-成都等主要商务航线）。
 - ◇ 选择分析维度：乘客身份证号归属地（或乘客登记的联系地址城市）。
 - ◇ 选择统计指标：乘客数量、占比。
 - ◇ 运行查询/生成报表。
 - ◇ 系统从乘客数据库和预订数据库中提取数据，进行聚合统计。
 - ◇ 结果以图表（如饼图、柱状图显示各城市乘客占比）和数据表格形式展示在界面上。
 - ◇ 刘女士查看分析结果，可以将报表导出为 Excel 或 PDF 格式，用于撰写分析报告提交给市场部。

✓ 结束： 数据分析师完成乘客客源地分布的查询与分析。

➤ 场景四：系统管理员为新入职票务代理创建账户

- ✓ 触发： 人力资源部通知 IT 部，新入职一位票务代理陈小姐，需要为其开通 APIMS 系统账户。
- ✓ 系统管理员操作：
 - ◇ 系统管理员赵工登录 APIMS 系统的管理后台。
 - ◇ 进入“用户账户管理”模块。

- ◇ 点击“创建新用户”按钮。
- ◇ 在表单中输入新用户的工号（作为登录名，如 T00123）、姓名“陈XX”、设置一个初始密码（并设定首次登录后强制修改）、选择用户角色为“票务代理”。
- ◇ 根据需要，还可以填写部门、联系方式等其他信息。
- ◇ 点击“保存”或“创建账户”。系统校验通过（如用户名未重复）后，创建新用户账户，并将其与“票务代理”角色关联，该角色已预定义了相应的操作权限。
- ◇ 赵工将账户信息（登录名、初始密码、系统访问地址）告知陈小姐，并提醒其首次登录后修改密码。
- ✓ 结束：系统管理员成功为新员工创建了系统账户并分配了角色权限。

这些场景覆盖了不同用户角色与 APIMS 系统的主要交互方式和业务流程，有助于理解系统的功能需求和使用环境。

2.4 系统性能指标规划

为了确保 APIMS 能够稳定、高效地支持航空公司的日常运营，需要设定明确的性能指标：

- 并发用户数 (Concurrent Users):
 - ✓ **核心内部用户（如票务代理进行 CRUD 操作、预订处理）：** 系统应能支持至少 50-150 个并发用户 同时进行操作，峰值期间（如节假日售票高峰、大规模促销活动）应能弹性扩展以应对更高的并发请求（例如通过增加服务器资源或优化负载均衡策略）。
 - ✓ **管理与分析用户（航班管理员、数据分析师、系统管理员）：** 并发数相对较低，预计在 10-30 个并发用户。
 - ✓ **外部乘客用户：** 初期目标支持 500-2000 个并发用户，并具备扩展到万级并发的能力。
- 总用户数 (Total Registered Users):
 - ✓ **内部员工用户：** 预计 200-500 名（包括所有票务代理、相关管理人员及 IT 人员）。
 - ✓ **注册乘客用户：** 长期目标可达到百万级别甚至更高。
- 核心业务响应时间 (Core Business Response Time):
 - ✓ **用户登录：** 平均响应时间应 < 1 秒。
 - ✓ **乘客信息查询（简单条件）：** 95%的查询结果应在 < 2 秒 内返回。

- ✓ 乘客信息列表加载（带分页，每页 20-50 条）： 首页加载时间应 <3 秒。
- ✓ 添加/修改乘客信息（保存操作）： 95%的操作应在 <3 秒 内完成并返回结果。
- ✓ 删除乘客信息： 95%的操作应在 <2 秒 内完成。
- ✓ 航班信息查询： 95%的查询结果应在 <3 秒 内返回。
- ✓ 创建机票预订： 95%的操作应在 <5 秒 内完成。
- ✓ 复杂数据查询/报表生成（数据分析师使用）： 响应时间根据数据量和查询复杂度而定，对于常规统计分析，应争取在 数秒到数十秒 内完成；对于大数据量离线分析，可接受更长时间。
- 数据容量与吞吐量：
 - ✓ 数据存储容量： 系统初期应能支持至少 1TB 的业务数据存储（包括乘客信息、预订记录、航班数据、日志等），并具备平滑扩展到 5-10TB 甚至 PB 级别 的能力。
 - ✓ 数据吞吐量： 在高峰期，系统应能处理每秒数百次的数据库读写请求（TPS/QPS），具体指标需结合详细业务量评估。
- 系统可用性 (Availability):
 - ✓ 核心业务功能模块(如乘客信息管理、预订处理)的可用性目标为 99.95% (年度非计划停机时间不超过约 4.38 小时)。
 - ✓ 对于非核心辅助功能，可用性目标可适当调整，但最低不应低于 99.9%。
 - ✓ 系统应具备故障快速恢复能力。
- 数据一致性与完整性：
 - ✓ 对于核心交易数据（如预订状态、支付状态），要求强一致性。
 - ✓ 系统必须确保所有数据的完整性，符合预定义的业务规则和约束（如主键唯一、外键参照、字段非空、格式正确等）。

2.5 系统战略地位、投资成本与预期收益

2.5.1 系统战略地位

APIMS 在航空公司的整体信息化战略中占据着至关重要的地位，其建设和成功运营将对公司产生深远的积极影响：

提升核心运营效率的基石： 乘客信息是航空公司所有面向客户业务的起点。APIMS 通过集中化、规范化管理乘客数据，能够显著提高票务处理、值机服务、客户咨询等核心运营环节的效率和准确性，降低人工操作成本和错误率。

增强客户关系管理 (CRM) 的数据引擎： 全面、准确、实时的乘客数据是构建有效 CRM 体系的基础。APIMS 能够为 CRM 系统提供高质量的数据输入，帮助航空公司更好地理解客户（如出行偏好、消费能力、忠诚度），从而实现精准营销、个性化服务和提升客户满意度与忠诚度。

数据驱动决策的关键支撑： 系统积累的运营数据（乘客构成、航线热度、预订趋势、客座率等）是宝贵的商业智能资源。通过对这些数据进行深入分析，可以为航线规划、运力调配、票价策略、市场推广等关键决策提供科学依据，提升决策的预见性和有效性。

推动数字化转型与服务创新的平台： APIMS 不仅是内部管理工具，更是航空公司向数字化、智能化服务转型的重要平台。在其基础上，可以逐步扩展在线自助服务（如网上订票、手机值机、行程管理）、常旅客计划的深度运营、以及与其他旅游服务提供商的合作，从而开辟新的收入来源，提升市场竞争力。

赢得客户与市场竞争优势的手段：

- ✓ **提升服务质量：** 通过快速准确的信息服务和个性化关怀，直接提升乘客的出行体验。
- ✓ **优化成本结构：** 通过自动化和效率提升，降低运营成本，从而可能在票价上获得竞争优势或提升盈利空间。
- ✓ **增强品牌忠诚度：** 优质服务 and 有效的常旅客计划有助于培养高价值的忠实客户群体。
- ✓ **快速响应市场：** 基于数据分析，能够更快地洞察市场变化并调整策略，抢占市场先机。

因此，APIMS 并非一个孤立的 IT 系统，而是支撑航空公司核心竞争力、驱动业务增长和服务创新的战略性基础设施。

2.5.2 建设投资与运营成本估算

建设 APIMS 需要一次性的初期投资和持续的年度运营成本。

➤ 初期投资估算 (一次性):

- ✓ **硬件采购与部署:** 包括应用服务器、数据库服务器、Web 服务器、网络设备、存储设备等。若采用云服务,则为初期的配置和部署费用。估算: ¥80,000 - ¥300,000。
- ✓ **软件采购与许可:** 操作系统、数据库管理系统 (but! MySQL 社区版免费,但商业支持可能收费)、中间件、开发工具、安全软件等。估算: ¥20,000 - ¥150,000。
- ✓ **系统开发与实施:** 包括需求分析、系统设计、编码、测试、集成、部署等所需的人力成本 (项目经理、架构师、开发工程师、测试工程师、UI/UX 设计师、DBA 等)。假设一个 5-8 人的团队,开发周期 6-12 个月。估算: ¥300,000 - ¥1,000,000+。
- ✓ **咨询与培训费用:** 对员工进行系统使用和维护培训的费用。估算: ¥10,000 - ¥50,000。
- ✓ **数据迁移成本:** 若存在大量历史数据需要从旧系统迁移到新系统,可能产生额外的数据清洗、转换和导入成本。估算: ¥10,000 - ¥80,000。
- ✓ **初期投资总估算范围:** ¥420,000 - ¥1,580,000+。

➤ 年度运营成本估算:

- ✓ **硬件维护与升级:** 服务器等硬件的年度维保费用、折旧、以及必要的升级替换。通常为硬件初投的 10-15%。
- ✓ **软件许可与支持续费:** 商业软件的年度许可费或订阅费、技术支持服务费。

- ✓ **IT 运维人力成本：** 系统管理员、DBA、应用维护工程师的部分薪资分摊。
- ✓ **服务器托管/云服务费用：** 若使用云平台，则为每年的服务租用费（根据资源使用量计费）。
- ✓ **网络与带宽费用。**
- ✓ **安全维护与审计费用。**
- ✓ **年度运营总估算范围：** ¥50,000 - ¥250,000+ 。

2.5.3 预期收益分析

APIMS 的预期收益可以从定量和定性两个方面进行分析：

➤ **可量化收益 (Quantitative Benefits):**

✓ **运营成本降低：**

- ✧ **人工成本节约：** 通过自动化处理，减少票务代理、客服等岗位在信息录入、查询、核对等方面的重复劳动时间，可能在不增加人手的情况下处理更多业务，或优化人员配置。
- ✧ **差错成本降低：** 减少因人工操作失误（如信息录错、重复预订）导致的退票、改签、赔偿等直接经济损失。
- ✧ **物料成本节约：** 推动无纸化办公，减少纸张、打印耗材等费用。

✓ **收入增加（间接或直接）：**

- ✧ **提升销售效率：** 更快的预订处理速度，可能抓住更多销售机会，特别是在销售高峰期。
- ✧ **精准营销带来的转化率提升：** 基于乘客数据分析，进行精准营销活动，提高营销投入回报率，带来额外销售收入。

➤ 难量化/定性收益 (Qualitative Benefits):

- ✧ 客户满意度提升： 更快、更准确、更个性化的服务体验，将显著提升乘客满意度。
- ✧ 客户忠诚度增强： 良好的服务体验和有效的常旅客计划运营（数据支持来自 APIMS）有助于培养忠实客户。
- ✧ 品牌形象提升： 高效、现代化的信息系统是航空公司专业形象的体现。

投资回报分析初步结论：

虽然 APIMS 的初期投资相对较高，但考虑到其在降低运营成本、提升服务质量、增强客户关系以及辅助决策方面的巨大潜力，预计系统上线后 1-3 年内即可通过运营成本的节约和效率的提升开始显现投资回报。长远来看，随着数据的积累和功能的深化应用（如精准营销、在线直销），其带来的战略价值和间接收益将远超初期投入。因此，从战略和经济角度综合考量，APIMS 项目具有较高的投资价值和良好的发展前景。

2.6 初步技术选型规划与技术可行性分析

2.6.1 初步技术选型规划

结合项目前期实验（Lab5）的实践以及当前主流技术趋势，APIMS 的初步技术选型规划如下：

➤ 后端技术栈：

- ✓ 编程语言： Python 3.10。Python 以其开发效率高、生态系统丰富、易于学习和维护等优点，非常适合 Web 应用和数据处理。
- ✓ Web 框架： Flask。Flask 是一个轻量级的 Python Web 框架，灵活性高，易于上手，适合构建 RESTful API 和中小型 Web 应用。对于 APIMS 的

核心功能，Flask 能够提供足够的支撑，并且可以根据需要方便地集成扩展。

- ✓ 数据库交互： `mysql-connector-python` (如 Lab5 所示)。

➤ 前端技术栈：

- ✓ 核心技术： HTML5, CSS3, JavaScript (ES6+)。这是现代 Web 开发的标准三件套。
- ✓ CSS 框架： Bootstrap 5 (如 Lab5 所示) 或其他流行的 CSS 框架 (如 Tailwind CSS, Ant Design 等，取决于 UI 设计偏好和团队熟悉度)。Bootstrap 能够快速构建响应式、美观的界面。
- ✓ API 通信： 原生 Workspace API。

➤ 数据库服务器：

- ✓ RDBMS： MySQL 8.3 。MySQL 在 Web 开发中应用广泛，与 Python 配合良好。PostgreSQL 在处理复杂查询和数据一致性方面有一定优势。

➤ API 设计风格：

- ✓ RESTful API。使用标准的 HTTP 方法 (GET, POST, PUT, DELETE) 对资源进行操作，数据交换格式采用 JSON。

➤ 架构模式：

- ✓ 三层/多层架构： 表示层 (前端)、应用层/业务逻辑层 (Flask 后端)、数据访问层 (数据库)。

2.6.2 技术可行性分析

综合评估以上技术选型，APIMS 在技术上是完全可行的，主要理由如下：

技术成熟度与稳定性： 所选技术（Python, Flask, MySQL/PostgreSQL, HTML/CSS/JS, Bootstrap, Nginx, Git, Docker）均为业界广泛应用、经过大规模实践检验的成熟技术。它们拥有庞大的开发者社区、丰富的文档资源和活跃的技术支持，这意味着在开发过程中遇到问题时更容易找到解决方案，技术风险较低。

技能匹配与学习曲线：

Python 语言以其简洁易学著称，Flask 框架也相对轻量，上手难度不高。项目前期实验（Lab5）已表明对 Python Flask 和 MySQL 有初步的实践经验。

可扩展性与可维护性：

Flask 的微框架特性使其具有良好的灵活性，便于根据业务发展添加新模块或集成第三方服务。

采用 RESTful API 设计，有利于实现前后端分离和微服务化改造（如果未来需要）。

遵循良好的编码规范、模块化设计和详细的文档，将有助于提高系统的长期可维护性。

容器化技术（Docker）简化了部署和环境管理，增强了系统的可移植性和可伸缩性。

3. 函数依赖、范式及 3NF 分解

3.1 函数依赖分析

在进行数据库设计时，函数依赖是用于确定一个表中属性之间关系的工具。它反映了一个属性（或属性集）对另一个属性（或属性集）的决定性关系。函数依赖的形式是：如果有关系 R 和属性集合 X 和 Y ，如果对于 R 中的每一对元组 t_1 和 t_2 ，只要 $t_1[X] = t_2[X]$ ，就必然有 $t_1[Y] = t_2[Y]$ ，那么就说 X 决定 Y ，

记作 $X \rightarrow Y$ 。

例如：乘客表中的函数依赖：

乘客 ID (passenger_id) \rightarrow 姓名 (name)、身份证号 (id_card_number)、电话号码 (phone_number)、电子邮件 (email)，即每个乘客的 ID 唯一地决定了其它信息。

3.2 范式与规范化

数据库规范化是为了消除冗余数据并防止潜在的更新异常。常见的规范化范式包括 1NF（第一范式）、2NF（第二范式）、3NF（第三范式）和 BCNF。

3.2.1 第一范式 (1NF)

要求每个字段只能包含原子值，表中的每一列都必须是不可再分的数据类型。确保每一列中的每个值都是单一的，且不可再分。例如：乘客表中的“联系方式”应拆分为多个列，如电话号码、电子邮件等，而不能放在一个字段中。

3.2.2 第二范式 (2NF)

满足 1NF，并且要求每个非主键字段完全依赖于主键。如果存在部分依赖（即部分依赖于主键的某些字段），则需要拆分成多个表。例如：如果有一个包含“航班 ID”和“机型”的表，且表中同时包含“航线”字段，由于“航线”仅依赖于“航班 ID”，而不是完整的复合主键，违反了 2NF。

3.2.3 第三范式 (3NF)

满足 2NF，并且要求没有传递依赖。即非主键字段不依赖于其他非主键字段。如果有字段依赖于非主键字段，则需要将其移到另一个表中。例如：在乘客表中，假设‘常旅客编号’依赖于乘客 ID，并且‘常旅客等级’依赖于‘常旅客编号’，这就违反了 3NF，因为‘常旅客等级’依赖于非主键字段‘常旅客编号’。

3.2.4 BCNF

满足 3NF，并且要求每个决定因素必须是候选键。如果表中有非候选键决定候选键的情况，则需要分解表。例如：如果表中有两个属性，且它们的组合决定了另一个属性，那么这些属性组合必须是候选键。

3.3 从不符合 3NF 的模式分解

通过对原始模式的分析，我发现表存在违反 3NF 的情况。具体的分解步骤如下：

➤ 分解 Flight 模式

原 Flight 模式存在的传递依赖：

- ✓ $\text{flight_id} \rightarrow \text{route_id} \rightarrow \text{departure_location}$
- ✓ $\text{flight_id} \rightarrow \text{route_id} \rightarrow \text{destination_location}$

分解后：

Flight_3NF(**flight_id**, airline_id, route_id, departure_time, arrival_time, total_seats, booked_seats, aircraft_model)

departure_location 和 destination_location 从该表中移除。

Route(**route_id**, airline_id, origin, destination, duration)

表保持不变，origin 对应起飞地点，destination 对应目的地点。查询时通过 Flight_3NF.route_id 与 Route.route_id 连接即可获取起飞和目的地点。

➤ 分解 Ticket 模式

原 Ticket 模式存在的传递依赖： $\text{ticket_id} \rightarrow \text{booking_id} \rightarrow \text{price}$

分解并调整后：

Ticket_3NF(**ticket_id**, booking_id, ticket_number, seat_number)

- ✓ booking_id 作为外键，引用 Booking 表的 booking_id。
- ✓ 移除了原有的 flight_id, passenger_id, 和 price 属性。这些信息都可以通过 booking_id 从 Booking 表中获得。

Booking(booking_id, passenger_id, flight_id, seat_type, booking_date, price, payment_status)

包含了票价信息。

3.4 变换后的 ER 图

