**集装箱装载优化系统项目需求分析**

**目录**

[集装箱装载优化系统项目需求分析 1](#_Toc6502)

[1. 引言 4](#_Toc14097)

[1.1 目的 4](#_Toc15947)

[1.2 背景 4](#_Toc14621)

[1.3 参考资料 4](#_Toc17301)

[1.4术语 4](#_Toc28176)

[2.任务概述 5](#_Toc8423)

[2.1目标 5](#_Toc447)

[2.2系统（或用户）的特点 5](#_Toc656)

[3.假定和约束 6](#_Toc10178)

[4.需求规定 6](#_Toc19498)

[4.1软件功能说明 6](#_Toc16935)

[4.2对功能的一般性规定 6](#_Toc6398)

[4.3对性能的一般性规定 6](#_Toc702)

[4.3.1 精度 7](#_Toc28832)

[4.3.2 时间特性要求 7](#_Toc965)

[4.3.3 灵活性 7](#_Toc22888)

[4.4 输入输出要求 7](#_Toc31486)

[4.5数据管理能力要求（针对软件系统） 7](#_Toc32686)

[4.6 故障处理要求 7](#_Toc632)

[4.7其他专门要求 7](#_Toc8184)

[4.8排拖要求 7](#_Toc3215)

[5. 运行环境规定 8](#_Toc4981)

[5.1 设备 8](#_Toc1260)

[5.2支撑软件 8](#_Toc6528)

[5.3接口 9](#_Toc28301)

[5.4控制 9](#_Toc3577)

[6. 尚需解决的问题 9](#_Toc3878)

[6.1尚需解决的问题 9](#_Toc31677)

[7.测试计划 9](#_Toc31311)

[7.1目的 9](#_Toc13187)

[7.2背景 9](#_Toc9053)

[7.3测试范围 9](#_Toc4830)

[7.4项目文件列表 10](#_Toc4831)

[8.测试需求 10](#_Toc17476)

[8.1分析各种信息 10](#_Toc1292)

[8.2需求组织成层次图 11](#_Toc18496)

[9.  测试策略 11](#_Toc8365)

[9.1 测试策略 12](#_Toc22842)

[9.2测试内容 12](#_Toc12869)

[10. 资源 12](#_Toc1660)

[10.1人力资源 12](#_Toc30084)

[10.2系统资源 13](#_Toc2136)

[11. 人员安排 13](#_Toc1864)

[11.1估计测试工作量 13](#_Toc7843)

[11.2创建工程调度表 13](#_Toc11452)

1. **引言**

**1.1 目的**

本报告旨在明确“集装箱装载优化系统”的功能需求、性能需求、运行环境及其他相关需求，为后续的系统设计、开发和测试提供依据。预期的读者包括项目经理、开发人员、测试人员及最终用户。

**1.2 背景**

系统名称：集装箱装载优化系统

行业情况：集装箱作为国际贸易中最主要的运输工具之一，其装载效率直接影响到物流成本和运输效率。当前市场上现有的集装箱装载优化系统存在功能单一、用户体验差、算法效率低等问题，无法满足用户对高效、智能装载方案的需求。

任务提出者：露营部

开发者：信息部

用户：露营部、OEM部

系统关系：本系统是一个独立的软件系统，旨在通过先进的算法和技术，提高集装箱的空间利用率，减少运输成本，并大幅降低人工排列的时间成本。

**1.3 参考资料**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 资料名称 | 简介 | 作者 | 日期 | 出版单位 |
|  |  |  |  |  |  |

|  |  |
| --- | --- |
| 网点 | 简介 |
|  |  |

**1.4术语**

打花垛：货物交叉排列，提高稳固性。

3D可视化：通过3D图形展示集装箱、货物、托盘的位置和形状。

蚁群优化算法（ACO）：一种基于蚂蚁觅食行为的优化算法，用于解决组合优化问题。

粒子群优化算法（PSO）：一种基于群体智能的优化算法，模拟鸟群或鱼群的行为。

模拟退火算法（SA）：一种基于物理退火过程的优化算法，用于寻找全局最优解。

**2.任务概述**

**2.1目标**

本系统的主要目标是通过先进的算法和技术，提高集装箱的空间利用率，减少运输成本，并大幅降低人工排列的时间成本。具体目标包括：

提高装载效率：通过智能算法实现货物的最优化排列，最大化集装箱的空间利用率，减少运输成本。

降低人工成本：减少人工排列货物的时间和错误率，提升工作效率。

增强货物稳固性：通过打花垛功能，确保货物在运输过程中的稳固性，减少货物损坏的风险。

提升用户体验：提供友好的用户交互界面、3D可视化、动态仿真和报表生成功能，帮助用户更好地理解和操作装载方案。

支持数据管理：通过数据库模块，实现货物信息、托盘信息、集装箱信息的高效存储和管理，确保数据的准确性和可追溯性。

提高系统性能：确保系统响应速度快、稳定性高、可扩展性强，满足使用者的操作需求。

**2.2系统（或用户）的特点**

系统特点：本系统采用先进的优化算法（如蚁群优化算法、粒子群优化算法和模拟退火算法），能够快速生成最优的装载方案。系统还提供3D可视化、动态仿真和报表生成功能，用户体验良好。

用户特点：系统的最终用户为露营部和OEM部，操作人员具备基本的计算机操作能力，能够通过图形用户界面（GUI）进行参数设置和结果查看。

**3.假定和约束**

假定：

算法有效性：假定蚁群优化算法（ACO）及其结合算法（PSO、SA）能够有效解决集装箱装载优化问题，并在合理时间内生成最优解。

用户需求稳定：假定用户需求在系统开发过程中不会发生重大变化，系统功能能够满足用户的实际需求。

数据准确性：假定用户提供的货物信息、托盘信息、集装箱信息等数据是准确且完整的。

约束：

开发期限：系统开发周期为4个月，试运行期为1个月，计划于2025年5月投入运行。

硬件性能：系统需要运行在支持Python环境的计算机上，建议配置至少8GB内存和多核处理器，以确保算法计算和3D渲染的性能。

**4.需求规定**

**4.1软件功能说明**

数据库模块：存储和管理货物信息、托盘信息、集装箱信息等。

装箱/打垛算法模块：实现货物的最优化排列，支持打花垛功能。

用户交互模块：提供3D可视化、动态仿真、报表生成和友好的用户界面。

**4.2对功能的一般性规定**

界面格式统一，统一的声音提示。

**4.3对性能的一般性规定**

**4.3.1 精度**

各种尺寸信息以毫米（mm）为单位，误差不超过0.5%

**4.3.2 时间特性要求**

系统应能够快速响应使用者的操作请求，数据库查询不超过2秒，算法计算不超过30秒。

**4.3.3 灵活性**

系统应具备良好的扩展性，能够适应未来的功能扩展或技术升级。

**4.4 输入输出要求**

输入：货物信息（尺寸、重量、类型）、托盘尺寸信息、集装箱尺寸信息、用户操作。

输出：3D可视化装载方案、动态仿真、报表生成。

**4.5数据管理能力要求（针对软件系统）**

数据量较小，暂无要求。

**4.6 故障处理要求**

系统应具备高可靠性和容错能力，确保长时间运行不出现故障。对于可能的软件、硬件故障，系统应能够自动恢复或提供相应的错误提示。

**4.7其他专门要求**

安全保密要求：系统应具备数据备份功能，保护用户数据不丢失、可找回。

使用方便的要求：系统应遵循软件工程敏捷性开发原则，保证用户交互友好。

**4.8排拖要求**

1. 货物的位置由尺寸、重量以及易变形程度共同决定，优先级为：易碎 > 重量 = 尺寸

2. 无论是散货还是打托，每个集装箱有一托货物需要包含每种品类的产品至少1箱，以打花垛的形式排列，且位置需要尽可能的靠近柜门，方便海关抽检

3. 所有的跺/托都需要打花垛

4. 整柜排柜的顺序需要从前往后一跺一跺的来进行，每跺的跨度不大于1200mm（纵向）

5. 托盘允许以横向、纵向的方式摆放

6. 散货不需要排拖，每跺的跨度不大于2310-23295mm（横向）

7. 海运排托，高度不超过2100（普柜）-2500mm（高柜/超高柜）

8. 海运散货不排托，高度不超过2250（普柜）-2260mm（高柜/超高柜）

9. 空运排托，高度不超过1500mm

10. 上下层货物之间允许存在悬空，但是悬空必须合理，能保证货物的稳定性

11. 货物摆放在托盘之上时，需在托盘四周各留有不小于10mm的距离

12. 托盘与托盘之间，前后需要留出20mm的间隙（纵向），左右需要留出100mm的间隙（横向）

13. 托盘在集装箱的前、左、右壁皆可靠边摆放，但是柜门出需留出不小于50mm的距离

1. **运行环境规定**

**5.1 设备**

硬件：8GB内存、多核处理器。

软件：Python环境、数据库管理系统。

**5.2支撑软件**

操作系统：Windows 10或更高版本。

数据库管理系统：SQLite。

其他支撑软件：PyQt5、VTK、SQLAlchemy、Pandas、manim、matplotlib.animation、jinja2、ReportLab、xlswriter。

**5.3接口**

如若需要，系统可设计通过API与其他物流管理系统进行数据交互，数据通信协议采用HTTP/HTTPS。

**5.4控制**

系统的运行通过用户界面进行控制，用户可以通过界面选择参数、调整摆放方案等。

1. **尚需解决的问题**

**6.1尚需解决的问题**

如何进一步优化算法，以缩短大规模数据计算时间。

如何提高3D渲染性能，确保在低配置计算机上也能流畅运行。

**7.测试计划**

**7.1目的**

本测试计划旨在确保“集装箱装载优化系统”的功能、性能和用户体验符合需求规格说明书的要求。

**7.2背景**

本系统是一个独立的软件系统，旨在通过先进的算法和技术，提高集装箱的空间利用率，减少运输成本，并大幅降低人工排列的时间成本。

**7.3测试范围**

测试范围包括功能测试、性能测试、用户界面测试、压力测试、容量测试、配置测试和安装测试。

**7.4项目文件列表**

|  |  |  |
| --- | --- | --- |
| 文档 | 已创建（是/否） | 版本/日期 |
| 可行性分析报告 | 是 | 1.0/2025-01-02 |
| 项目开发计划 | 是 | 1.0/2025-01-04 |
| 软件需求说明书 | 是 | 1.0/2025-01-06 |
| 概要设计说明书 | 否 |  |
| 详细设计说明书 | 否 |  |
| 测试计划 | 否 |  |
| 测试分析报告 | 否 |  |
| 项目开发总结报告 | 否 |  |
| 软件问题报告 | 否 |  |
| 软件修改报告 | 否 |  |
| 源程序 | 否 |  |
| 用户手册 | 否 |  |

**8.测试需求**

**8.1分析各种信息**

反复检查并理解各种信息，和用户交流，理解他们的要求。可以按照以下步骤执行：

1）确定软件提供的主要商业任务：集装箱装载优化

2）确定完成该任务所要进行的交易：货物信息输入、装载方案计算、3D可视化展示

3）确定从数据库信息引出的计算结果：最优装载方案

4）确定所要的时间和条件：数据库查询不超过2秒，算法计算不超过30秒

5）确定应用需要处理的数据量：货物信息、托盘信息、集装箱信息

6）确定需要的软件和硬件配置：Python环境、8GB内存、多核处理器

7）确定安装过程：定制安装、升级安装

8）确定没有隐含在功能测试中的户界面要求：操作与显示的一致性、界面遵从合理标准

**8.2需求组织成层次图**

提高集装箱装载效率

├──────── 数据库管理

│ ├── 货物信息存储

│ ├── 托盘信息存储

│ └── 集装箱信息存储

├──────── 装箱/打垛算法

│ ├── 货物排列优化

│ └── 打花垛功能

└──────── 用户交互

├── 3D可视化

├── 动态仿真

└── 报表生成

1. **测试策略**

**9.1 测试策略**

|  |  |
| --- | --- |
| 测试策略项 | 例 |
| 测试阶段 | 系统测试 |
| 测试类型 | 功能测试 |
| 测试技术 | 75% 用SQA Suite自动测试，25% 手工测试 |
| 完成标准 | 95% 测试用例通过并且最高级缺陷全部解决 |
| 特殊考虑 | 无 |

**9.2测试内容**

1）功能测试：验证系统功能是否符合需求规格说明书的要求

2）用户界面测试：验证用户界面是否友好、易用

3）性能测试：验证系统响应时间、算法计算时间是否符合要求

4）压力测试：无

5）容量测试：无

6）配置测试：验证系统在不同硬件配置下的运行情况

7）安装测试：验证系统的安装过程是否顺利

1. **资源**

**10.1人力资源**

|  |  |  |
| --- | --- | --- |
| 职位 | 姓名 | 特殊责任/说明 |
| 测试经理 |  |  |
| 测试工程师  设计/开发（可以多人） |  |  |
| 测试工程师  测试执行（可以多人） |  |  |
| 测试系统管理员 |  |  |

**10.2系统资源**

|  |  |
| --- | --- |
| 系统 | 名称/类型 |
| 硬件环境  软件环境  专门配置要求  客户测试机  其他要求 | 8GB内存、多核处理器 |
| Python环境、SQLite数据库 |
| 无 |
| 无 |
| 无 |

1. **人员安排**

**11.1估计测试工作量**

测试设计：10天

测试开发：15天

测试执行：20天

测试评估：5天

**11.2创建工程调度表**

|  |  |
| --- | --- |
| **任务** | **相关工作量（天）** |
| **测试计划** | **8** |
| 确定项目 | 1 |
| 定义测试策略 | 2 |
| 决定测试需求 | 3 |
| 生成测试计划文档 | 2 |
| **测试设计** | **7** |
| 分析测试需求 | 2 |
| 指定测试过程 | 1 |
| 指定测试用例 | 2 |
| 查看测试需求的覆盖率 | 2 |
| **测试开发** | **14** |
| 建立测试开发环境 | 3 |
| 开发测试过程 | 5 |
| 测试和调试测试过程 | 2 |
| 修改测试过程 | 2 |
| 重新测试并调试测试过程 | 2 |
| **测试执行** | **8** |
| 设置测试系统 | 2 |
| 执行测试 | 3 |
| 验证测试结果 | 2 |
| 生成缺陷日记 | 1 |
| **测试评估** | **4** |
| 回顾测试日记 | 1 |
| 评估测试需求的覆盖率 | 1 |
| 评估缺陷 | 1 |
| 决定是否达到测试完成的标准 | 1 |