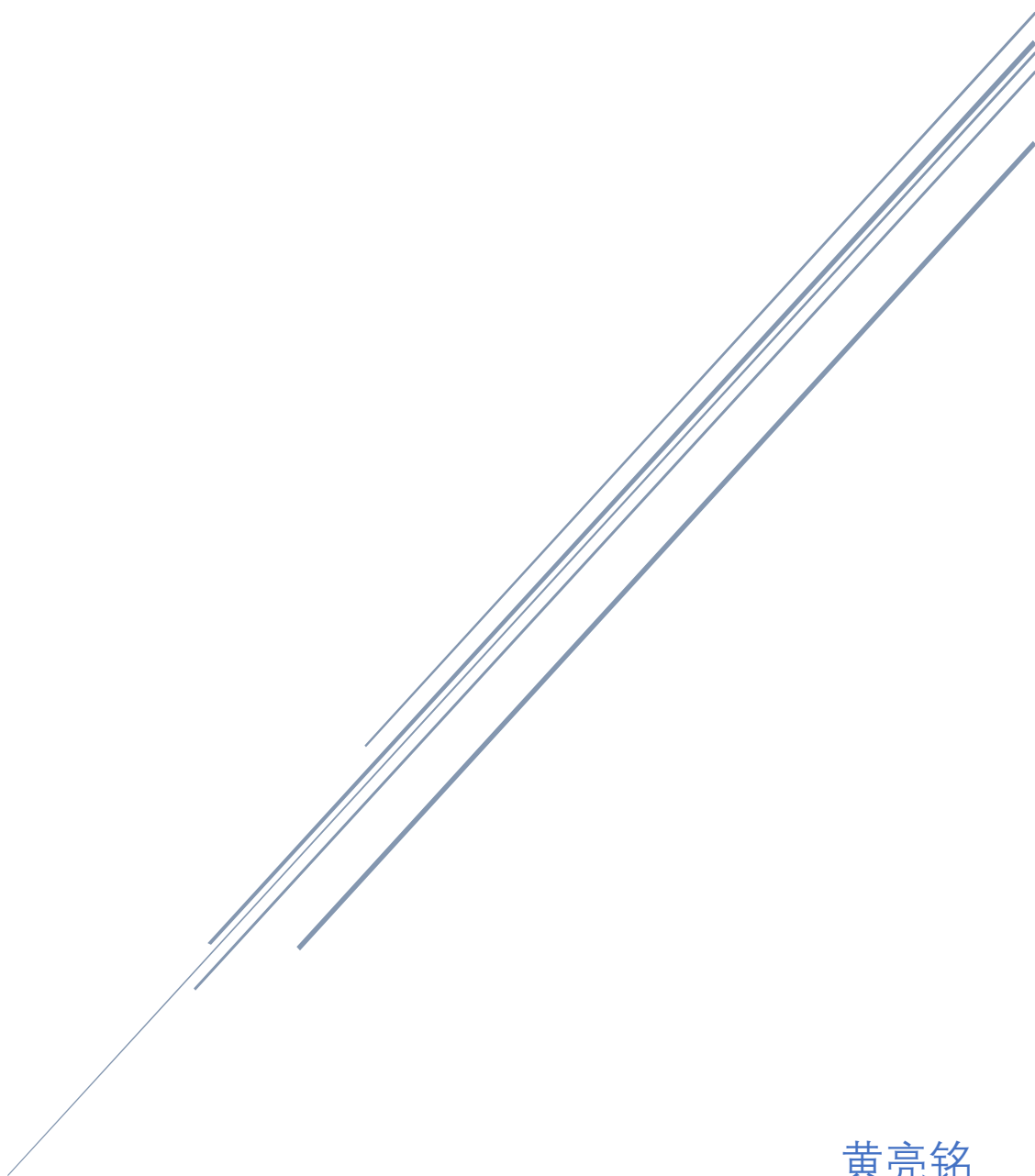


账易通

可行性分析报告



黄亮铭

2022155028

目录

1 项目背景	2
1.1 用户痛点	2
1.2 市场潜力	2
2 项目目标	2
2.1 基础功能目标	2
2.2 用户体验目标	3
2.3 长期目标	3
3 可能的解决方案	3
3.1 前端	3
3.2 后端	4
3.3 数据库	4
3.4 评估	4
4 可行性分析	4
4.1 技术可行性	4
4.2 经济可行性	5
4.3 操作可行性	5
4.4 社会可行性	6
5 开发计划	6
6 需求分析	7
6.1 用户需求	7
6.2 市场需求	7
6.3 社会需求	7

1 项目背景

在当今社会,随着个人消费和理财意识的增强,越来越多的用户开始关注日常收支管理,期望通过简单的记账方式来实现财务状况的可视化。然而,市面上的许多记账应用要么功能复杂、操作繁琐,要么带有过多广告。这些因素导致部分用户在使用时体验不佳,无法有效、便捷地完成记账和财务管理。

1.1 用户痛点

- **操作复杂**: 大部分记账软件内置了大量功能,如分类管理、自动化记账等,但这些功能对一些基础用户来说过于复杂,反而加大了使用难度。
- **广告过多**: 许多应用嵌入了广告,甚至在每次记账时弹出,使用户体验变差。
- **付费要求**: 部分应用只允许使用部分基础功能,需要付费订阅才能解锁所有基础功能和高级功能,这对于希望只使用基础功能的用户来说并不划算。
- **缺乏个性化分析**: 许多用户希望记账应用不仅能记录数据,还能提供基本的财务分析,以帮助自己了解收支趋势,实现合理的支出规划。

1.2 市场潜力

根据调查,日常记账软件的用户数量逐年增长,尤其是年轻一代的用户更倾向于使用移动应用来管理财务。同时,许多用户偏好“简洁”和“直观”功能,渴望一种无需深入学习、即可快速上手的记账工具。因此,市场上存在着对**简单高效、无广告、数据安全**的记账工具的需求空缺。

2 项目目标

本项目的目标是开发一款简洁无广告、功能全面和操作便捷的记账小程序,为用户提供一种高效管理日常收支的工具,帮助用户轻松管理个人财务,实现收支记录、预算设置、财务分析等功能。

本项目希望能通过上述功能成为用户日常生活中不可或缺的财务管理工具,帮助用户实现财务自由,提升生活质量。

2.1 基础功能目标

- **收支记录**: 用户能够快速记录收入和支出。支持分类标签(如餐饮、交通、购物等),并允许用户自定义分类,方便追踪具体消费类别。

- **预算提醒**：允许用户设定月度或年度预算，当支出接近或超出预算时，系统会自动提醒，帮助用户控制消费。
- **数据统计与可视化分析**：提供基础的数据统计功能，包括月度、年度收支概览。通过柱状图、饼图等可视化形式，帮助用户清晰了解消费结构。
- **数据导出**：支持将记账数据导出为 Excel 文件，便于用户备份或用于其他用途。

2.2 用户体验目标

- **简洁直观的用户界面**：设计简洁易用的界面，减少用户操作步骤，让用户能够快速上手、轻松完成记账操作。
- **无广告**：应用内无任何广告，确保用户能够专注于记账体验，不被无关信息干扰。
- **全面且可靠的功能**：提供全面的财务管理功能，包括但不限于收支记录、预算管理、等。

2.3 长期目标

- **用户行为分析**：基于匿名化的用户数据分析，为用户提供更智能的财务建议，如消费优化提示、储蓄目标提醒等。
- **个性化建议功能**：基于用户的消费习惯，提供个性化的储蓄和支出管理建议，帮助用户更好地规划财务。

3 可能的解决方案

本项目采用分层架构，分为前端，后端和数据层。前端使用 Uniapp，后端使用 Go，数据层使用 MySQL。

3.1 前端

前端负责数据展示与用户交互，通过 HTTP 请求与后端通信以获取和提交用户数据。使用 Uniapp 进行前端开发拥有如下优势：

1. **跨平台开发**：Uniapp 是一款基于 Vue.js 的跨平台框架，支持一次编写、多端部署，兼容 IOS 和 Android，非常适合构建多平台应用。
2. **丰富的组件和插件**：Uniapp 拥有广泛的组件库和插件市场，能够快速实现表单输入、图表展示等常用功能，节省开发时间。
3. **开发和调试效率高**：支持灵活的实时调试与热更新功能，能够加快开发和调试速度。

3.2 后端

后端作为 API 服务层, 接收前端请求, 进行数据处理和业务逻辑实现, 并与数据库交互。

使用 Golang 进行后端开发具有如下优势:

1. **高性能与简洁性:** Go 语言编译快、执行效率高, 代码简洁、开发效率高, 适合构建高性能、稳定的后端服务。
2. **丰富的生态和库支持:** Go 生态中有丰富的库和框架, 支持快速构建 Web API 服务, 简化开发流程。

3.3 数据库

数据库的作用是存储和管理用户数据, 包括账单记录和分类信息, 并提供数据读写接口。

数据库选择方面目前暂定为 MySQL。使用 MySQL 作为数据进行开发具有以下优势:

1. **稳定可靠:** MySQL 是一个成熟的关系型数据库, 性能稳定, 适合存储结构化的财务数据, 支持数据查询优化。
2. **数据一致性与安全性:** 支持 ACID 特性, 能保证事务一致性, 适合记录用户的财务数据。

3.4 评估

使用 Uniapp、Go 和 MySQL 的技术组合, 可以有效地实现跨平台、高性能、数据安全的记账小程序。该解决方案不仅具备良好的可扩展性, 还能够保障数据的安全性和高效访问, 满足用户日常记账和财务分析需求。

4 可行性分析

本项目的可行性分析主要从技术可行性、经济可行性、操作可行性和社会可行性四个方面进行分析。

4.1 技术可行性

技术可行性方面主要从资源分析和效率分析两方面进行探讨。

4.1.1 资源分析

本项目采用 Uniapp、Go 和 MySQL 作为技术栈，具备充足的开发资源和技术支持。

- Uniapp 基于 Vue.js，支持跨平台快速开发，兼容移动端和小程序平台，提供丰富的组件库和 API 接口，适合构建轻量的财务管理工具。
- Go 语言可以通过 Gin 框架快速构建 API 服务。
- MySQL 则用于结构化数据存储，支持事务处理，保证数据一致性和可靠性，适合记账应用的数据需求，支持的索引优化技术还能够提升查询性能。

4.1.2 效率分析

从开发效率方面来看，Uniapp 支持热更新和调试，能够快速进行 UI 调整；Go 语言的简洁语法和快速编译特性减少了代码开发和调试时间；MySQL 作为成熟的数据库，配合 GORM 等 ORM 框架，使得数据操作简便快捷，减少了数据管理的复杂性。三者相结合可以显著提升开发效率。

从运行效率方面来看，Go 语言在并发处理上表现出色，能高效处理 API 请求，确保系统响应速度。此外，MySQL 数据库优化能够在大数据量下保持高效的数据查询和统计分析。

4.2 经济可行性

本项目的成本可以分为两个方面，一方面是前期开发的成本，另一方面是后期运营的成本。

从前期的开发成本的角度来看，本项目开发人员可控，采用开源框架进行开发，降低了框架授权和维护成本。由于采用跨平台开发，节省了单独开发不同平台应用的资源和时间，进一步降低了总体开发成本。

从后期运营成本的角度来看，记账小程序的数据存储主要在本地数据库中，如需云端备份，仅需轻量级存储服务即可，较少涉及高成本的云计算服务，运营成本较低。

本项目的收益主要分为两种，一种来自高级功能如 AI 分析等的订阅收益，另一种是软件打开时广告的分成收益。

虽然本项目的经济收益来源单一，但是考虑到技术为市场上的开源技术、人力资源成本较低和硬件需求较低的方面，本项目在未来仍然有一定的盈利空间。

4.3 操作可行性

1. **项目开发周期：**根据项目的规模和功能需求，项目可分阶段开发和测试，确保在有限时间内完成基础功能的开发和优化。使用的技术框架（Uniapp、Go、MySQL）都有丰富的社区支持和开发文档，能够帮助开发团队快速解决技术问题，保证开发效率。

2. **后期维护与升级**: 项目架构灵活, 采用前后端分离, 便于后期功能扩展和维护升级。未来版本中, 可在后端增加新的 API 接口, 同时前端可通过无缝更新直接获取新功能, 进一步提升操作体验。

4.4 社会可行性

1. **理财意识的普及**: 记账应用作为个人财务管理的入口产品, 可以在一定程度上帮助用户培养理财意识, 提升用户的财务管理能力。符合社会倡导的理性消费和经济规划的理念, 具备一定的社会推广价值。
2. **用户需求契合度**: 随着个人理财意识的增强, 市场上对便捷、无广告的记账应用需求逐渐增多。许多用户在日常生活中希望快速管理支出并获得数据分析支持, 帮助自己掌控财务状况。该记账小程序的简洁设计符合用户习惯, 满足对轻量化财务管理工具的需求。

5 开发计划

本项目的工期要求为 5 周, 我将 5 周时间分配如下所示。

1. **需求分析（第一阶段）（1 周）**: 首先通过调研市面上存在的记账小程序, 整理出应用的核心功能。然后编写需求说明文档, 包括功能描述、用户故事和使用场景, 确保对项目需求有清晰的定义。最后根据需求的紧迫性和用户需求频率对功能优先级进行排序, 将核心功能优先开发
2. **系统设计（第二阶段）（1 周）**: 制定项目的整体架构, 明确前后端分离模式, 定义前端与后端的交互接口, 以及数据库的存储模式。
3. **前后端开发（第三阶段）（2 周）**: 前端实现收入/支出添加、预算设置、数据展示和数据导出等核心页面; 后端构建 API 服务, 实现增删改查接口, 以处理前端的收支记录、查询历史记录等请求。
4. **测试（第三阶段）（1 周）**: 使用不同的测试方法, 如单元测试、集成测试和系统测试等, 对程序进行完备的测试, 确保程序各个模块均能以符合预期的方式运行。
5. **维护（第四阶段）（持续）**: 通过用户反馈、评论等渠道收集使用体验和建议, 整理改进需求, 为后续更新提供依据。同时对用户反馈的 Bug 和问题进行定期维护和修复, 并持续优化性能, 确保应用的稳定性。

6 需求分析

6.1 用户需求

1. **简洁的记账流程**: 用户希望能够快速、方便地记录日常收支, 避免复杂的操作步骤, 适应日常使用中的高频率需求。简洁的界面和流程可以帮助用户在最短时间内完成记账, 降低使用门槛。
2. **预算提醒**: 用户希望可以设定月度或年度预算, 且在支出接近或超出预算时, 系统能发出提醒, 以便控制消费, 达到省钱的目标。
3. **数据统计与可视化**: 用户希望查看月度、年度的收支分析, 以柱状图、饼图等形式呈现数据, 帮助用户直观地了解消费结构和趋势, 便于财务规划。
4. **离线使用**: 用户希望即使在没有网络的情况下也能完成基本的记账操作。

6.2 市场需求

1. **轻量级记账应用的空缺**: 市场上现有的记账软件大多具有丰富的功能, 但其复杂度和高频广告影响了用户体验。许多用户希望拥有一款简洁、无广告的记账应用, 因此存在针对轻量级记账工具的市场需求。
2. **个性化与扩展性**: 随着用户理财意识的增强, 记账应用不仅需满足基本的记录功能, 还应具备个性化的扩展能力, 例如消费分析、趋势预测等, 以吸引不同类型用户的长期使用。

6.3 社会需求

1. **普及财务管理意识**: 在消费和金融普及的时代, 越来越多的年轻人、家庭需要掌控收支状况, 提高理财意识。通过提供便捷的记账方式, 帮助用户了解自己的消费行为, 有助于社会整体的消费观念向理性化、科学化转变。
2. **支持经济规划和储蓄**: 一些用户不仅希望了解自己的消费情况, 还希望应用能够在预算管理、储蓄目标等方面提供支持, 以实现长远的经济规划目标。便捷的记账应用可以帮助用户提升财务管理能力, 从而提升生活质量。