**基于flutter和kotlin ktor框架博客系统的设计与实现**

摘　要

随着移动互联网的发展和智能手机普及，移动端博客成为人们学习和交流的主要途径和工具。利用 Flutter 框架开发的博客系统，为用户提供移动客户端博客发布与交流平台，利用评论和功能，建立起社区文化。解决移动客户端编辑发布文章的需求。

博客系统使用Flutter作为前端框架，采用 MVVM 架构。本博客系统主要包括博文管理、用户关系管理、收藏和关注、评论管理等功能模块。利用Flutter框架的跨平台能力，同时适配Android、IOS平台。以Markdown的形式发布文章，使用 flutter\_markdown插件进行文本到Markdown效果转换，使用dio插件进行网络请求，使用provider和 rxdart 插件进行状态管理。前端利用Material Design理念，实现应用对全局主题和主题模式的自定义，并支持语言的选择。后端使用Kotlin Ktor作为开发框架，数据库采用ORM框架，建立数据库表与关系，通过DAO层对数据库进行操作。

本博客系统实现了对文章内容、文章评论、用户详情的浏览功能，用户登录注册功能，登录后编辑、删除、收藏文章，登录后关注用户，登录后添加删除评论等功能。经测试功能稳定，功能具有可拓展性，响应迅速。

**关键词：**Flutter，MVVM，Kotlin Ktor，ORM

**Design and Implementation of Blog System Based on Flutter and Kotlin Ktor Framework**

Abstract

With the development of mobile Internet and popularization of smart phones, mobile blogs have become the main way and tool for people to learn and communicate. The blog system developed by the Flutter framework provides users with a mobile client blog publishing and communication platform, and uses comments and functions to build a community culture. Solve the needs of mobile client editing and publishing articles.

The blog system uses Flutter as the front-end framework and uses the MVVM architecture. This blog system mainly includes functional modules such as blog post management, user relationship management, collection and attention, and comment management. Take advantage of the cross-platform capabilities of the Flutter framework and adapt to both Android and IOS platforms. Post articles in the form of Markdown, use the flutter\_markdown plugin for text-to-Markdown effect conversion, use the dio plugin for network requests, and use the provider and rxdart plugins for state management. The front end uses the Material Design concept to realize the customization of the global theme and theme mode by the application, and supports the choice of language. The back end uses Kotlin Ktor as the development framework, the database uses the ORM framework, establishes database tables and relationships, and operates the database through the DAO layer.

This blog system implements the browsing function of article content, article comments, user details, user login registration function, edit, delete, collect articles after login, follow users after login, add delete comments after login and other functions. The tested function is stable, the function is expandable, and the response is rapid.

**Key Words:** Flutter, MVVM, Kotlin Ktor, ORM

**目　录**

1. 绪 论

1.1 课题背景

在这个网络时代，人们可以通过博客系统记录自己日常发生的事情或自己感兴趣的兴趣爱好，与他人分享自己的感受，交流自己所学的知识，探讨感兴趣的话题与问题。同时结识志趣相投的朋友，建立起文化社区。同时，也有越来越多的专业知识的博客出现，使之成为学习和交流的重要工具之一。 随着智能手机的普及，移动客户端成为主流，逐渐代替PC客户端和网页版客户端。目前，移动客户端的博客系统相对较少，功能相对简单。大多数没有编辑和发布文章的能力；文章管理的体验不友好，用户不能随时随地发布最新的文章，不能及时关注最新的文章动态；该博客系统主要解决用户的此类问题。

1.2 意义

利用Flutter开发的博客系统，为用户提供在移动客户端编辑和管理文章功能。并且提供评论和收藏功能，让用户更方便的发现和专注于自己感兴趣的内容，及时获取更新。同时可以关注自己喜欢的用户，以及查看自己的粉丝，方便找到志同道合的朋友。 文章采用markdown作为文章编辑语言，以方便简洁的纯文本快速高效的编辑出样式丰富的博客内容。且存储代价小。 同时利用Flutter框架的跨平台能力，在一次性开发的情况下，同时适配Android、IOS等多个平台，为不同平台用户带来一致的软件体验。

1.3 主要内容

本博客系统主要包括博文管理、用户管理、收藏和关注、评论管理等功能模块。使用“flutter” 作为前端框架，“kotlin ktor”作为后端框架。 前端采用 mvvm 架构，高效的响应UI请求并依据数据更新UI。数据库采用ORM架构，方便数据库操作与管理。 所有请求采用http协议的get请求方式

1.3.1 博文管理

用户登录后可以编辑、发布新文章，查看自己的全部文章，删除过时的文章。

1.3.2 收藏和关注

用户登录后可以收藏或取消收藏自己喜欢的文章，关注其他用户，以及查看被哪些用户关注。

1.3.3 用户管理

用户可通过注册获得新账号。登录后可修改用户名和头像，修改头像。

1.3.4 评论管理：

用户登录后可对文章进行评论，删除其他人对自己文章的评论。删除自己发表的评论。回复评论。

1.3.5 基础设置

客户端支持自定义主题，选择客户端内使用的语言。

1. 关键技术介绍

2.1 Flutter 框架

Flutter是 Google 的一套开源的UI工具包，具有跨平台能力，在一套代码库的情况下，高效构建多平台精美应用，支持移动、Web、桌面和嵌入式等众多平台。Flutter框架开源、免费，拥有宽松的开源协议。拥有诸多优势，Flutter的热重载功能可以快速地进行测试、构建UI、添加功能并更快地修复错误。在iOS和Android模拟器或真机上可以在亚秒内重载，并且不会丢失状态。同时，Flutter内置美丽的Material Design和Cupertino（iOS风格）widget、丰富的motion API、平滑而自然的滑动效果和平台感知，为用户带来全新体验。 Flutter拥有丰富的工具和库，有助于轻松地同时在iOS和Android系统中实现想法和创意。以轻松快捷的方式来构建漂亮的移动应用程序。Flutter支持混合开发，可与现有项目并存，复用现有的Java、Kotlin、Swift等代码，访问iOS和Android上的原生系统功能和系统SDK。使用Flutter的现代、响应式框架，和一系列基础widget，轻松构建用户界面。使用功能强大且灵活的API（针对2D、动画、手势、效果等）解决艰难的UI挑战。

2.2 Kotlin 语言

Kotlin 语言是运行于Java虚拟机的静态类型语言，语法简洁偏向于函数式，大大减少样板代码的编写，高效开发。避免空指针异常等整个类的错误，最大化保证程序安全。同时可以复用JVM的现有库，生态强大。Kotlin 原生支持协程，于1.3版本宣布协程达到稳定。Kotlin/Native可将Kotlin代码编译为无需虚拟机就可运行的原生二进制文件，该技术是基于LLVM的Kotlin标准库的原生实现，并根据不同平台的特点，在内存管理等方面进行差异化处理，使得性能更优。 Kotlin 适用于服务端、Android、移动端跨平台、数据科学、Web开发等多种场景，并于2017年Google I/O开发者大会成为Android开发的一等语言。

2.3 Ktor 框架

Ktor 是一个以Kotlin作为开发语言构建异步服务器与客户端的框架，主要由服务器引擎和灵活的异步客户端两部分组成，基础框架轻量，将所有附加功能插件化，作为可选项。同时添加和配置功能方便易懂。客户端凭借Kotlin丰富的跨平台库，支持JVM，JS，Android和iOS等多平台，在跨平台移动应用程序上广泛应用。该博客系统中添加Location插件，用于获取请求中的参数，添加Gson插件便于返回Json格式的数据包。

2.4 Exposed库

Exposed 是用Kotlin 编写的轻量的SQL库，主要包括两层数据库访问：类型安全的SQL包装DSL和轻量级数据访问（DAO）。支持MySQL、MariaDB、SQLite、Oracle、SQL Server等多款主流数据库。后端采用Exposed库实现数据库的ORM框架，将数据库结构映射为Kotlin代码，提高开发效率，降低开发成本，更加安全。同时便于数据库迁移以及加入缓存等功能。Exposed 库底层充分利用Kotlin协程，提高处理效率。

2.5 MVVM架构

MVVM是Model-View-ViewModel 的简称，是前端主流的高效响应框架。View层负责主要UI的构建，将业务逻辑抽离到ViewModel中，两者实现数据层绑定；同时ViewModel可与Model通信，在Model处理完请求后，ViewModel通知View进行响应。MVVM架构具有低耦合、可重用性、独立开发、可测试的优势。

1. 需求分析

3.1 功能分析

系统功能层次图如图3.1所示。



图3.1 系统功能层次图

3.1.1 浏览内容

博客系统设计为弱登录模式，在未登录的情况下，用户只有浏览文章及评论，查看作者信息，个性化app的权限。无法添加、删除、收藏文章，无法关注用户，无法添加、删除评论。

3.1.2 注册

用户注册时必须提供用户名、邮箱、登录密码，其中用户名与邮箱不准与其他用户已使用的重复；可选提供头像图片的链接地址、性别。默认图片来源为空，默认性别为保密。注册成功后自动登录

3.1.3 登录

用户登录时可选择用户名或邮箱作为账号，提供登录密码。用户登录成功后生成随机身份鉴权码用于用户身份校验。并返回给前端保存，用户在进行浏览以外的权限操作时，需传递身份鉴权码，后端验证身份有效性后允许操作，若身份鉴权码过期，需重新登录后获取

3.1.4 文章管理

登录后，用户可编辑自己已有文章或编辑新文章。文章标题不能为空，文章内容使用Markdown语法进行编辑，提交前用户可以随时进行预览。提交后，后端添加或更新文章内容。每篇文章记录创建时间与最新更新时间。

用户可删除个人文章，删除文章同时，将取消其他用户对文章的收藏，删除文章所有评论。

用户可查看自己已发布的文章，可进行查看详情、编辑文章、删除文章、删除文章相关评论的操作

3.1.5 收藏文章

用户可收藏或取消收藏他人的文章，不能收藏自己的文章。可浏览已收藏文章，可选择进行查看详情或取消关注操作

3.1.6 关注用户

用户可关注或取消关注其他用户，用户不能关注自己。用户可浏览已关注的用户，拥有查看该用户详情或取消关注的操作。

用户可查看自己的粉丝，可以进行查看粉丝详情和关注粉丝的操作

每个用户的详情包括用户基本信息，发布的文章，收藏的文章，关注的用户，粉丝和发布的评论等信息。

3.1.7 评论

用户可对任意文章添加评论。用户可浏览已发布的评论，并进行查看被评论文章和删除操作

3.2 前端技术可行性分析

前端使用flutter开发，采用MVVM架构，逻辑层与视图层尽力分离。同时对于频繁使用的数据进行缓存处理，减少对服务器的请求。请求提交服务器之前，对数据合法性进行简单检测，避免无效请求。对于请求失败返回的信息进行适当处理和显示。

1. 系统设计