**ICyclist 项目说明文档**

**一、项目概述**

ICyclist 是一个基于 Android 平台的骑行社交应用，采用 Java 语言开发，主要面向骑行爱好者，为其提供社交互动平台。

**二、核心功能模块**

1. 用户认证模块

登录界面：支持用户ID/密码登录

自动登录：可选功能，默认关闭

登录状态管理：使用 SharedPreferences 持久化存储

2. 主界面架构

四大核心界面：骑友圈、社区社区、发布界面、个人中心

动态 Fragment 切换：单 Activity 多 Fragment 架构

底部导航：四个主要功能入口

3. 社交功能模块

骑友圈（微博式动态）：帖子列表展示、支持点赞、评论互动、下拉刷新功能、实时内容更新

兴趣社区：社区分类浏览模拟社区数据展示可扩展的社区详情

内容发布：纯文本帖子发布、发布后自动更新骑友圈、防重复提交机制

个人中心：用户信息展示、数据统计（帖子数、粉丝、关注）、退出登录功能

**三、技术架构**

**1、数据模型：java**

**User.java** —— 用户数据模型

作用：定义用户数据结构，支持序列化传输

// 关键属性

private String userId; // 用户唯一标识

private String username; // 显示名称

private String password; // 登录密码

private String bio; // 个人简介

private int postCount; // 发帖数统计

private int followers; // 粉丝数

private int following; // 关注数

public User(String userId, String username, String password) *// 构造函数*

**Post.java** - 帖子数据模型

作用：定义帖子内容结构和互动数据

// 关键属性

private String postId; // 帖子唯一标识

private String userId; // 发布者

IDprivate String username; // 发布者名称

private String content; // 帖子内容

private long timestamp; // 发布时间戳

private int likeCount; // 点赞数

private int commentCount; // 评论数

private boolean isLiked; // 当前用户是否点赞

private String community; // 所属社区

public Post(String postId, String userId, String username, String content) *// 构造函数*

**用户认证模块**

LoginActivity.java - 登录控制

作用：处理用户认证流程和自动登录逻辑

关键方法：

initUserData() - 初始化模拟用户数据

validateUser(String userId, String password) - 用户凭证验证

saveLoginStatus(User user) - 保存自动登录状态

checkAutoLoginStatus() - 检查并执行自动登录

**主界面容器**

MainActivity.java - 应用主控制器

作用：管理应用整体架构和界面切换

关键属性：

private BottomNavigationView bottomNavigationView; // 底部导航private FrameLayout fragmentContainer; // 内容容器private List<Post> postList; // 全局帖子数据private User currentUser; // 当前用户信息

核心方法：

**界面管理方法：**

showFeedFragment() - 显示骑友圈

showCommunityFragment() - 显示兴趣社区

showPublishFragment() - 显示发布界面

showProfileFragment() - 显示个人中心

**数据操作方法：**

setupFeedRecyclerView(View feedView) - 初始化帖子列表

loadMockData() - 加载模拟帖子数据

updateFeedAfterPublish() - 发布后立即更新显示

**界面适配器类**

PostAdapter.java - 帖子列表适配器

作用：管理帖子项的显示和交互逻辑

关键方法：

onBindViewHolder() - 绑定数据到视图

updateLikeButton(boolean isLiked) - 更新点赞状态

getTimeAgo(long timestamp) - 时间格式化显示

CommunityAdapter.java - 社区列表适配器

作用：显示兴趣社区列表和基本信息

关键方法：

onBindViewHolder() - 绑定社区数据

**2、核心组件**

Activity：LoginActivity, MainActivity

Fragment：动态切换的四个界面

Adapter：PostAdapter, CommunityAdapter

布局：Material Design 风格

**3、关键技术点**

视图管理：Fragment 动态加载与切换

数据绑定：RecyclerView + Adapter 模式

状态持久化：SharedPreferences 存储登录状态

线程安全：Handler 处理 UI 更新

用户体验：下拉刷新、防重复点击

**项目结构**

app/src/main/java/com/example/icyclist/

├── LoginActivity.java # 登录界面

├── MainActivity.java # 主界面容器

├── User.java # 用户数据模型

├── Post.java # 帖子数据模型

├── PostAdapter.java # 帖子列表适配器

└── CommunityAdapter.java # 社区列表适配器

res/layout/

├── activity\_login.xml # 登录界面布局

├── activity\_main.xml # 主界面布局

├── fragment\_feed.xml # 骑友圈布局

├── fragment\_community.xml # 兴趣社区布局

├── fragment\_publish.xml # 发布界面布局

├── fragment\_profile.xml # 个人中心布局

├── item\_post.xml # 帖子项布局

└── item\_community.xml # 社区项布局

**四、扩展建议**

添加图片上传功，实现真实的评论系统，增加帖子删除功能，实时消息推送，骑行轨迹记录等

**注意事项**

当前使用模拟数据，无网络请求层

使用测试账号：user001 / 123456