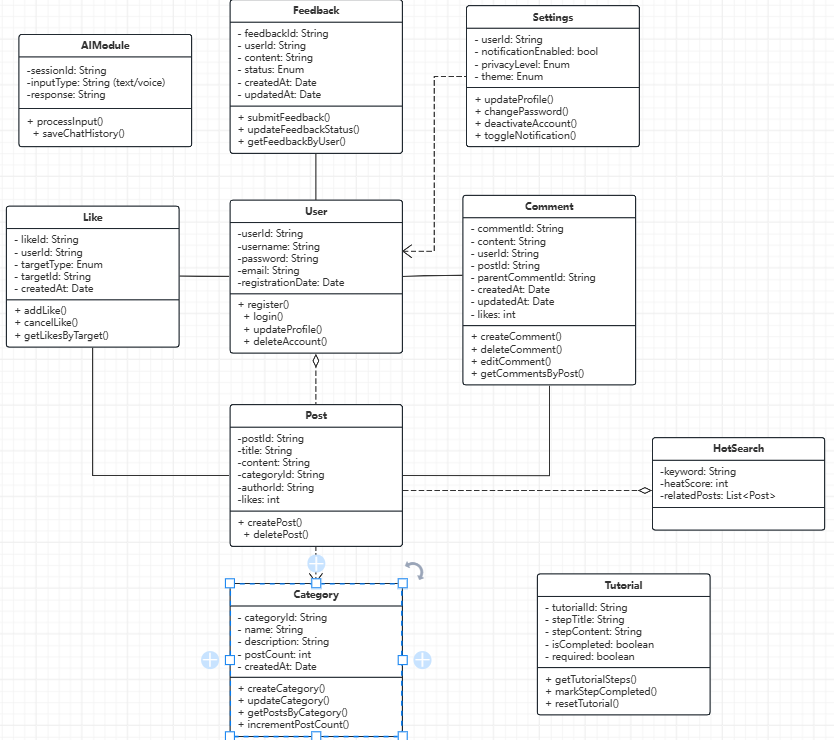


这张用例图展示了校务问答机器人的主要功能模块及用户与之的交互关系。用户可通过认证模块进行注册、登录和退出登录操作，实现身份验证；在发帖与互动模块，能进行发帖、评论、点赞、搜索等操作，满足交流互动需求；AI 模块支持用户与 AI 交互并查看聊天记录，获取校务相关解答；反馈与帮助模块允许用户提交反馈，获取新手引导；设置模块则提供修改昵称、账号、密码以及注销账号等个性化设置功能 。这些模块协同工作，为全体师生提供便捷的校务问答及互动服务。

该类图呈现了校务问答机器人系统的关键类及其属性与方法 ，描述如下：

**用户相关**

**User 类**：代表用户，涵盖用户 ID、用户名、密码等基础信息属性。具备注册、登录、更新个人资料、删除账户等方法，用于实现用户在系统中的身份管理与信息维护。

**内容发布与互动相关**

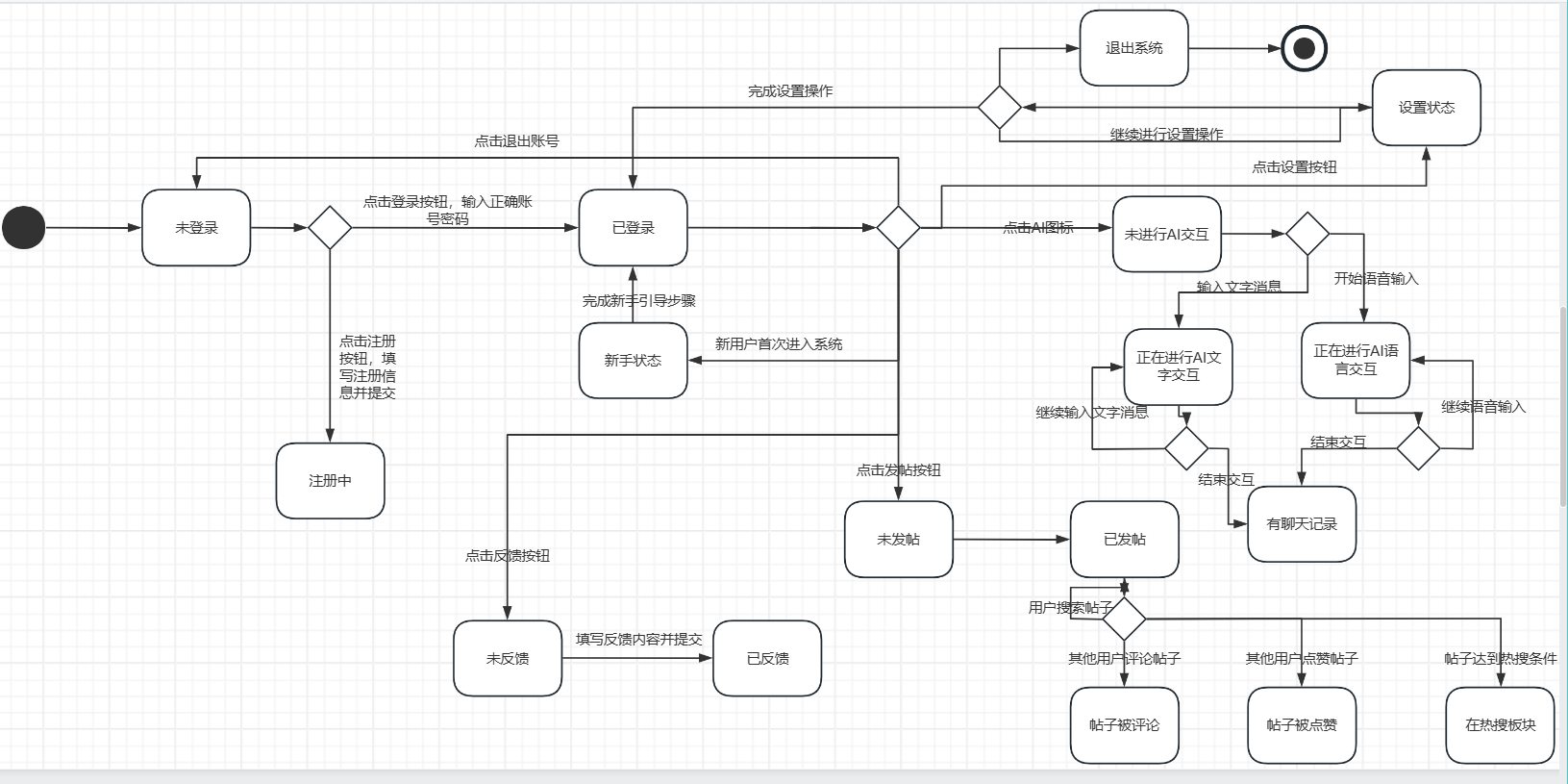
* **Post 类**：对应发布的帖子，有帖子 ID 、标题、内容等属性。支持创建帖子、删除帖子等操作。
* **Comment 类**：用于评论，包含评论 ID 、内容等属性，可进行创建、删除、编辑评论及按帖子获取评论等操作。
* **Like 类**：处理点赞功能，记录点赞 ID 等信息，能执行添加点赞、取消点赞、按目标获取点赞等操作。

**系统功能相关**

* **AIModule 类**：AI 模块，负责处理用户输入（支持文本 / 语音 ）并生成响应，还可保存聊天记录。
* **Feedback 类**：反馈模块，记录反馈 ID 等信息，支持提交反馈、更新反馈状态、按用户获取反馈等操作。
* **Settings 类**：设置模块，涉及用户设置项，如用户名、通知开关等，可进行更新个人资料、修改密码等操作。

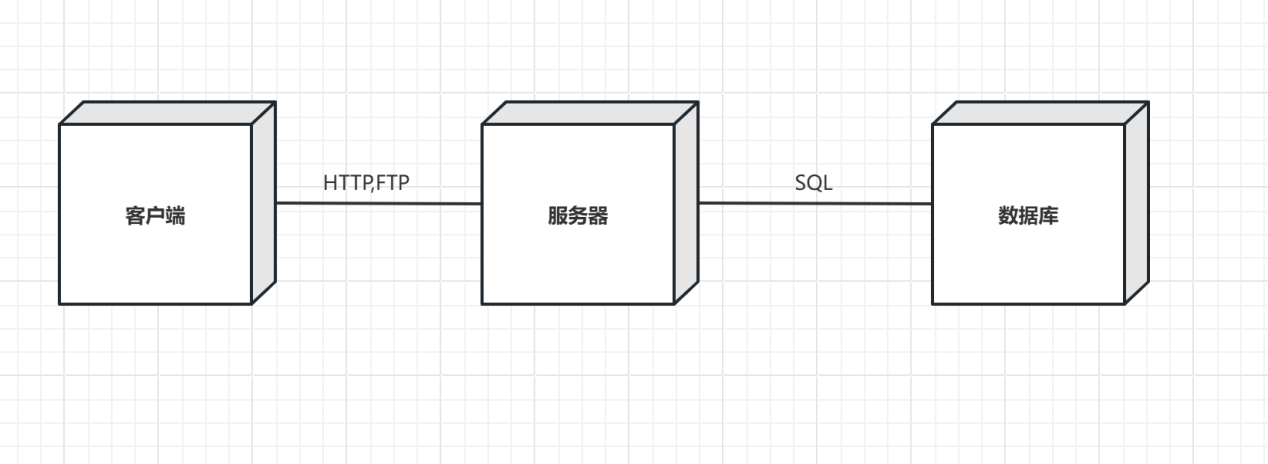
**辅助与分类相关**

* **Category 类**：用于帖子分类，有分类 ID 等属性，可进行创建、更新分类及按分类获取帖子等操作。
* **Tutorial 类**：新手引导模块，包含引导步骤 ID 等属性，可获取引导步骤、标记完成状态等。
* **HotSearch 类**：热搜模块，记录关键词、热度得分等，关联相关帖子 。 这些类相互协作，支撑起校务问答机器人的各项功能。



这张状态图描述了校务问答机器人系统中用户的不同状态及其转换流程：

* **未登录状态**：用户初始处于此状态。可点击注册按钮进入注册中状态，填写注册信息并提交完成注册；也可点击登录按钮，输入正确账号密码进入已登录状态。
* **已登录状态**：新用户首次进入系统为新手状态，完成新手引导步骤后脱离新手状态。在此状态下，点击反馈按钮，填写反馈内容并提交可从未反馈转为已反馈；点击发帖按钮从未发帖转为已发帖；点击设置按钮进入设置状态，完成设置操作后可选择继续操作或退出系统。
* **设置状态**：可继续进行设置操作，或点击退出账号回到未登录状态。
* **AI 交互相关状态**：点击 AI 图标，未进行 AI 交互时，可选择输入文本消息开始文字交互，或开始语音输入进行语音交互。交互过程中，根据输入情况持续交互，结束交互后，若有聊天记录则保留相关记录。
* **发帖相关状态**：已发帖状态下，帖子可能因用户搜索、其他用户评论或点赞，满足条件时进入热搜模块。



1.客户端层

用户直接操作的终端界面（如Web浏览器、移动应用等）

通过标准协议（HTTP/FTP）与服务器通信

2.服务器层

接收客户端请求并处理业务逻辑

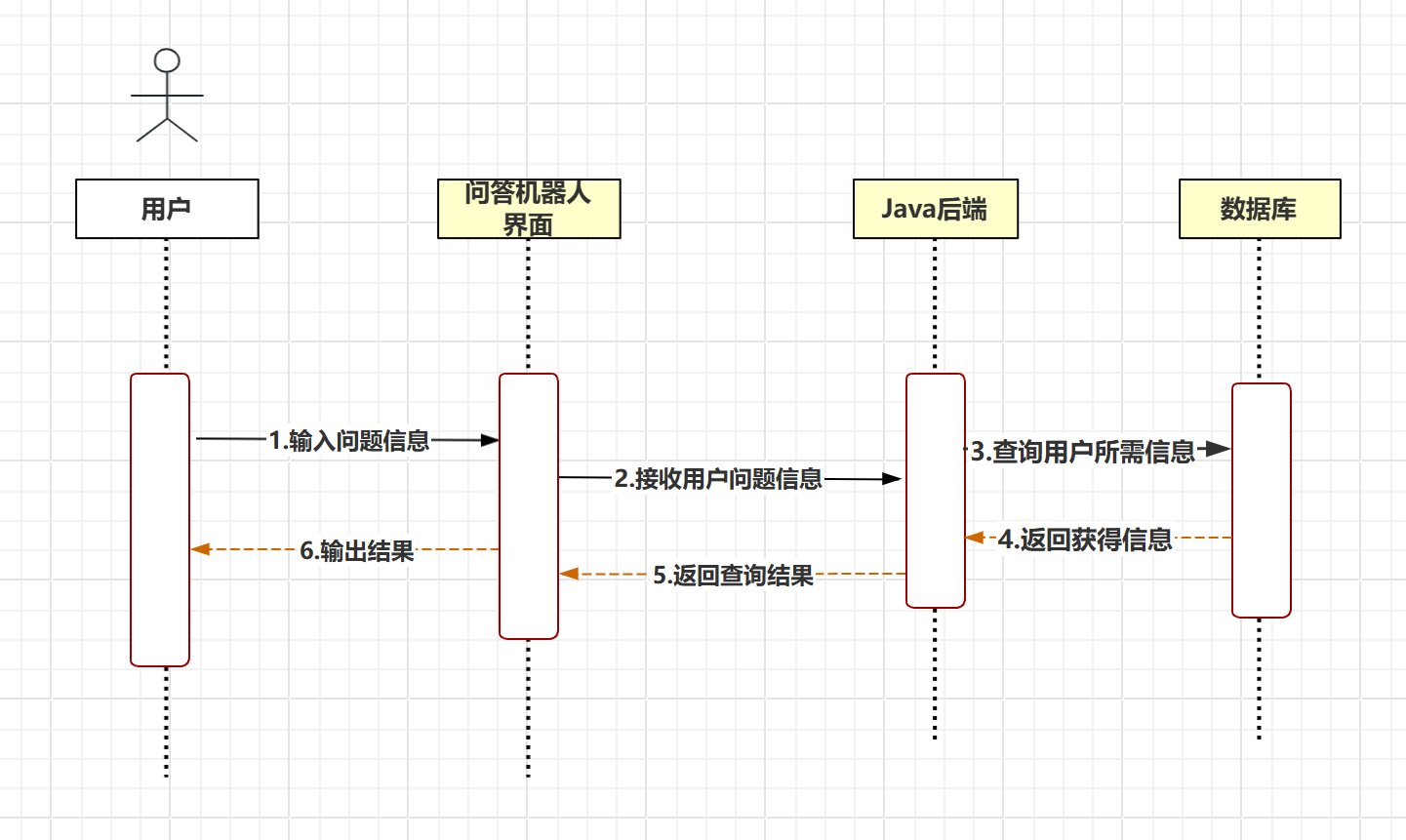
作为中间层协调前后端数据交互

3.数据库层

使用SQL语言进行数据存储与管理

持久化保存系统核心数据

数据流向：  
客户端 →（通过HTTP/FTP协议）→ 服务器 →（通过SQL查询）→ 数据库



该图清晰展示了智能问答机器人系统的完整交互流程，描述了用户从提问到获得答案的数据流转过程。各模块功能及交互逻辑如下：

1. 用户层

通过问答界面（如小程序、网页等）输入问题

最终接收并查看机器人返回的答案

2. 问答机器人界面

作为前端交互入口，收集用户问题（步骤1）

将问题传输至Java后端（步骤2）

向用户可视化展示最终结果（步骤6）

3. Java后端

接收并解析用户问题（步骤2）

生成数据库查询逻辑（步骤3）

将数据库返回的原始数据加工为结构化答案（步骤5）

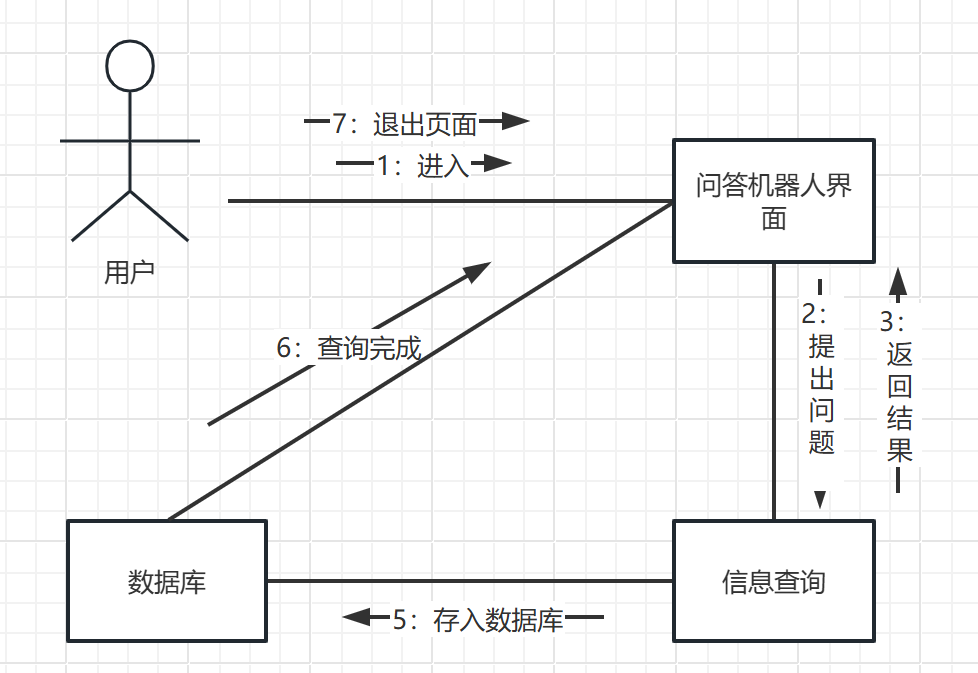
4. 数据库层

存储知识库和用户数据

执行SQL查询并返回匹配信息（步骤4）

核心数据流：

用户输入 → 界面传输 → 后端处理 → 数据库查询 → 结果返回 → 界面呈现



本图展示了智能问答机器人系统的完整闭环交互流程，采用UML序列图风格呈现了用户与系统各组件之间的动态交互过程。以下是关键环节解析：

1. 用户端交互流程

初始操作：用户通过"1：进入"动作启动问答界面

核心交互：通过"2：提出问题"发起查询请求

终止操作：通过"7：退出页面"结束会话

2. 系统处理流程

界面层：接收用户问题并触发"信息查询"指令

数据层：完成"5：存入数据库"的持久化操作

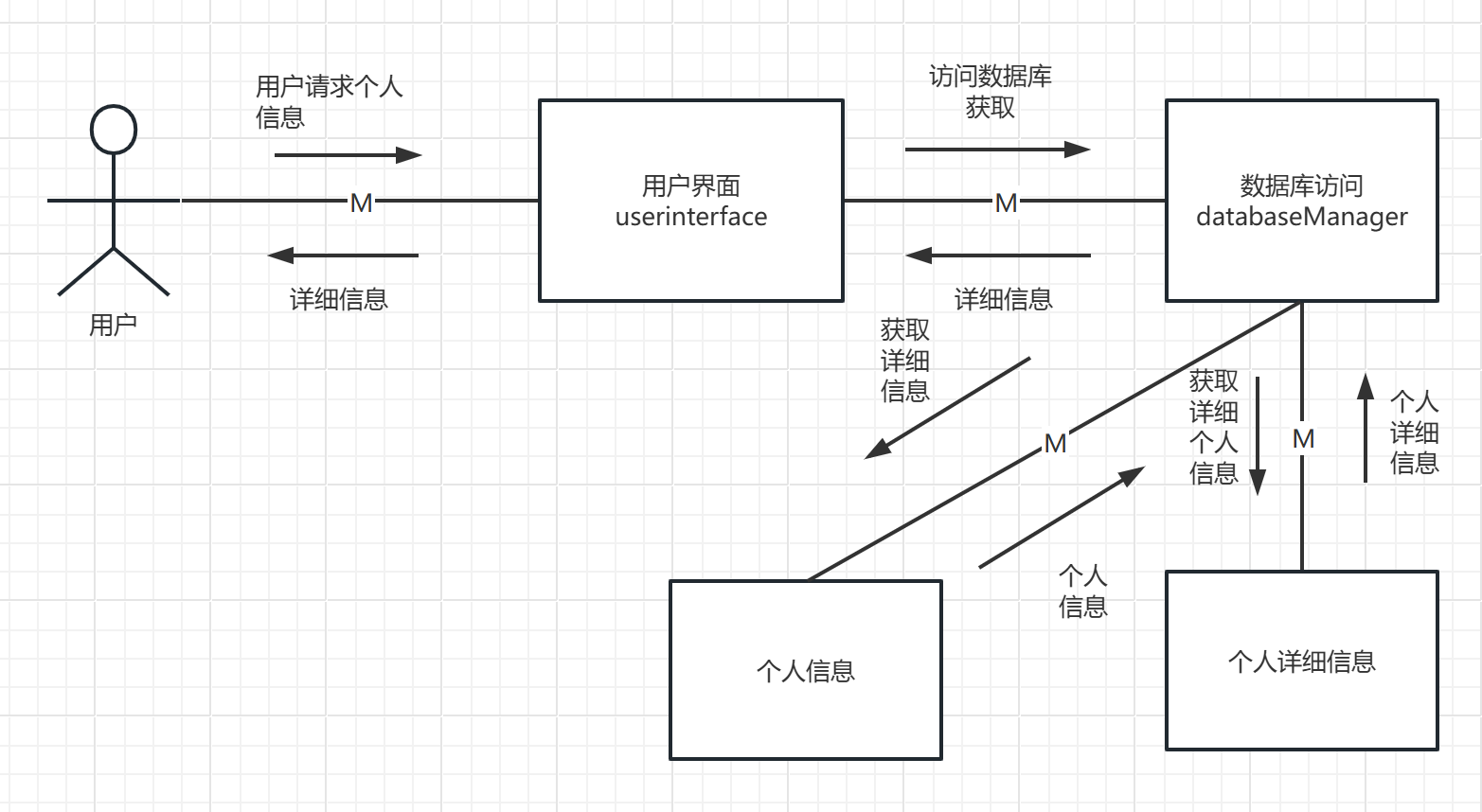
响应闭环：通过"3：返回结果"和"6：查询完成"形成完整应答周期

3. 数据流向特征

用户请求流：用户 → 界面 → 数据库

系统响应流：数据库 → 界面 → 用户

完整闭环：包含查询发起、执行、存储、反馈全环节



本图详细描述了用户查询个人信息的完整系统交互过程，展示了从用户请求到数据返回的端到端处理流程：

1. 核心交互流程

用户发起"用户请求个人信息"指令

用户界面接收请求并触发"获取个人信息"操作

数据访问层执行"访问数据库获取"动作

系统最终向用户返回"详细信息"

2. 组件职责划分

用户层：发起请求并接收最终数据展示

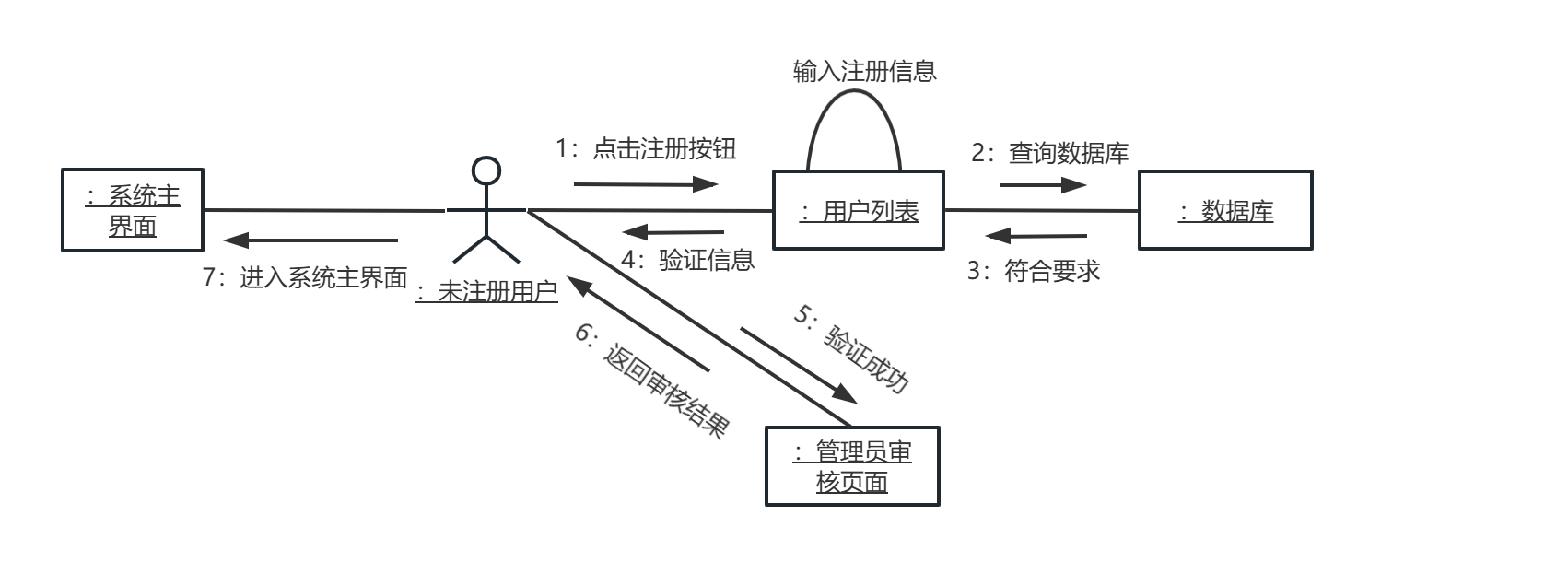
界面层：作为中转处理单元，连接前后端

数据层：负责与数据库的实际交互

3. 数据流转特征

请求流：用户 → userinterface → databaseManager → 数据库

响应流：数据库 → databaseManager → userinterface → 用户



用户注册及登录验证系统流程图说明

1. 注册阶段流程

用户初始操作："输入注册信息"并"1：点击注册按钮"

系统后台处理：

"2：查询数据库"进行信息核验

"3：符合要求"时完成注册

2. 登录验证阶段

用户进入："输入系统主界面"

安全验证：

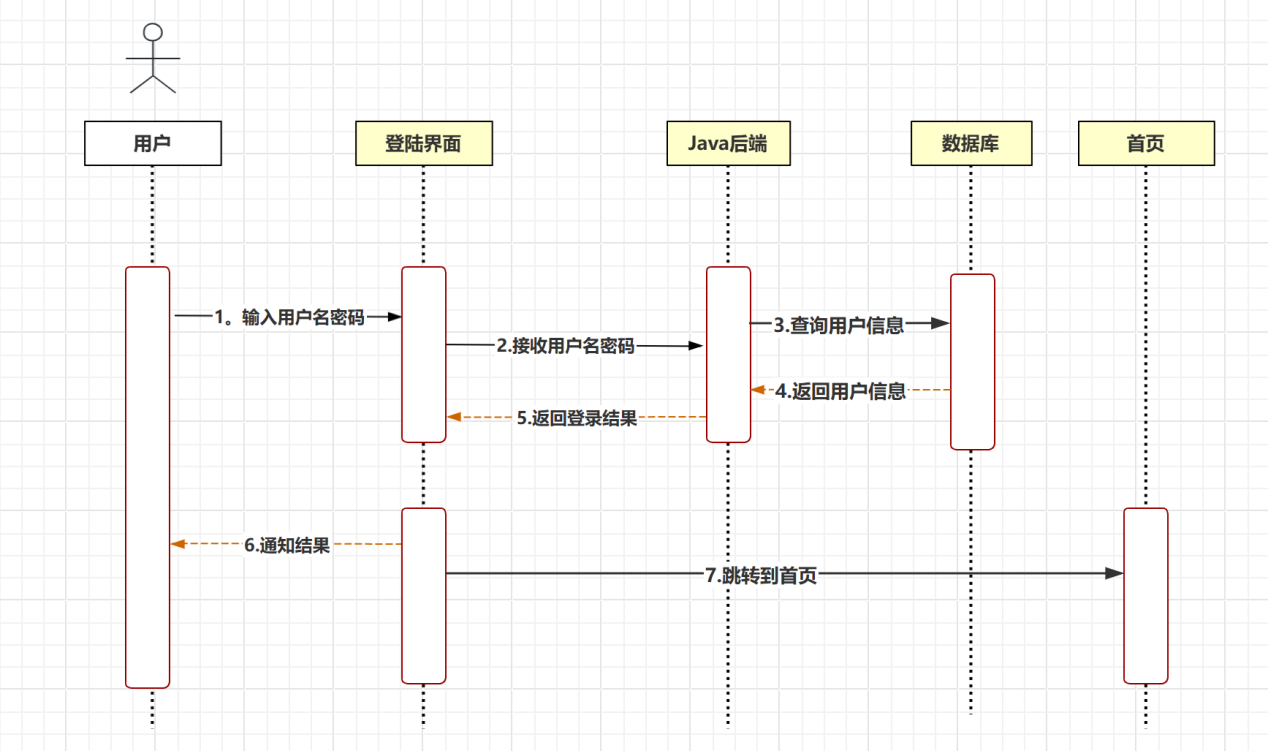
"4：验证信息"（通常包括账号密码验证）

"5：验证成功"后授予访问权限

3. 界面跳转逻辑

成功路径："7：进入系统主界面"

备用路径："6：返回商城登录"（处理验证失败或中断情况）



本图详细描述了应用中典型的用户登录验证流程，展示了从用户输入凭证到成功跳转首页的完整数据交互过程：

1. 核心登录流程

用户操作阶段：

"1. 输入用户名密码"（在登陆界面）

系统验证阶段：

"2. 接收用户名密码"（Java后端）

"3. 查询用户信息"（数据库验证）

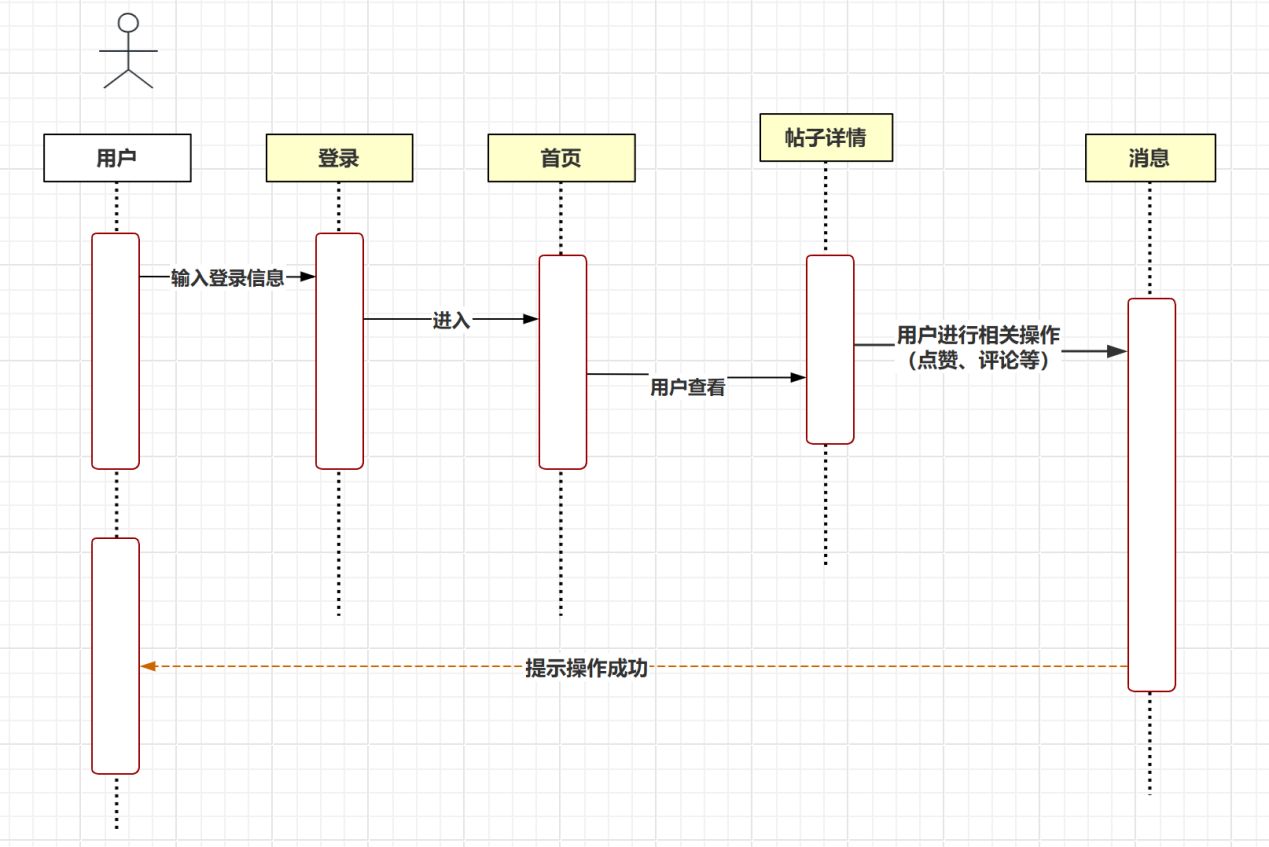
"4. 返回用户信息"（数据库→后端）

结果反馈阶段：

"5. 返回登录结果"（后端处理）

"6. 通知结果"（界面展示）

"7. 跳转到首页"（成功时）



本图清晰展示了用户的典型使用路径，从登录到内容互动的完整操作流程：

1. 核心用户流程

登录阶段：

"输入登录信息" → "进入"（完成身份认证）

内容消费阶段：

"首页"浏览 → "帖子详情"查看

互动参与阶段：

"用户进行相关操作"（点赞、评论等社交行为）

系统反馈"提示操作成功"

2. 界面跳转逻辑

线性路径：登录 → 首页 → 帖子详情 → 消息

操作闭环：每个互动动作都伴随系统确认反馈