**GoMuseum 产品架构设计**

## 架构亮点：

1. **技术栈选择**：
   1. 前端采用Flutter实现跨平台，配合Riverpod状态管理
   2. 后端采用微服务架构，可选Node.js或Python FastAPI
   3. 使用向量数据库支持图像识别检索
2. **清晰的代码组织**：
   1. 采用Clean Architecture分层设计
   2. 功能模块化，便于团队协作开发
   3. 每个模块都有完整的data/domain/presentation三层
3. **核心功能实现**：
   1. 详细的识别流程（在线/离线双模式）
   2. 完整的语音交互流程
   3. 智能的离线包管理策略
   4. 灵活的付费系统
4. **性能优化**：
   1. 客户端图片压缩和预加载
   2. 服务端缓存和批处理
   3. 智能的离线包同步机制
5. **可扩展性**：
   1. 微服务架构便于横向扩展
   2. 清晰的API设计
   3. 完善的监控和分析体系

## 使用建议：

1. **分阶段实现**：
   1. Phase 1: 先实现识别、讲解、基础UI
   2. Phase 2: 加入离线包、支付、AI问答
   3. Phase 3: 优化性能、增加社交功能
2. **与Claude Code协作**：
   1. 可以将架构文档分模块输入给Claude Code
   2. 先实现核心识别流程，验证技术可行性
   3. 逐步完善其他功能模块
3. **关键技术点**：
   1. 图像识别可以先用现成的API（如Google Vision、阿里云视觉）
   2. TTS可以使用云服务，后期再考虑自建
   3. 离线包可以先做简单版本，后续优化压缩算法

这个架构设计已经足够详细，可以直接开始编码实现。建议你先从识别模块开始，这是整个产品的核心。有任何具体实现问题都可以继续问我！