# 数据摄取与 PDF 解析（服务端 + Celery 队列）

目标：将 PDF/教材转换为结构化的段落、标题、注释与图片占位，为后续个性化与生成提供稳定输入。能力在服务端实现；Flutter 客户端负责文件上传、状态展示与结果查询。

## 队列编排与任务划分

* 任务列表：
* ingest\_pdf(file)：存储文件、提取文本/版式元素，输出 PageBlock[]
* clean\_and\_chunk(blocks)：清洗与分块，输出 Chunk[]
* embed\_chunks(chunks)：生成向量，写入向量库
* index\_chunks(chunks)：将分块与元数据入库（便于回溯）
* 状态机：PENDING -> STARTED -> RETRY -> FAILURE | SUCCESS
* 进度上报：维护 stage（upload/parse/clean/chunk/embed/index）与 progress(0-100)，GET 轮询返回
* 重试策略：tenacity/backoff 指数退避，对 OCR/模型/存储等外部依赖失败时有限重试

## 实现与提示词

### 1) 上传与导入（移动端透传）

* 客户端：dio multipart → POST /ingest/pdf → 返回 taskId → 轮询 GET /ingest/tasks/:id
* 测试（接口）：

curl -F file=@sample.pdf http://localhost:3001/ingest/pdf | jq .
curl http://localhost:3001/ingest/tasks/<taskId> | jq .

* 测试（客户端）：integration\_test 验证状态迁移与弱网重试

### 2) 解析 PDF（服务端）

* 工具：pdfminer.six/pymupdf（Python）或 pdf-parse（Node）
* 约定输出：PageBlock{ pageNumber, blocks:[{type:'heading'|'paragraph'|'list', text, bbox}] }[]
* 单测：多版式样例，断言页数、标题层级、断行合并

### 3) 清洗与分块

* 策略：去页眉页脚、合并断行、按语义/长度分块（token 友好）
* 接口：cleanAndChunk(blocks, {targetTokens: 400, overlap: 50})
* 单测：边界条件（空页、表格、脚注）与 Chunk 边界

### 4) 向量化与检索

* 起步：pgvector；进阶：Qdrant/MongoDB Atlas Vector
* 接口：embedChunks(chunks) / search(query, topK)
* 单测：固定 query 与人工标注 Top-K 一致性

### 5) 图片与 OCR（可选）

* 工具：tesseract、PaddleOCR
* 单测：关键词召回/相似度阈值

## 验收

* 3 份不同排版 PDF 的解析、清洗、分块稳定；
* 单测>95% 通过，关键函数边界覆盖充分；
* 客户端上传/轮询流程稳定，弱网/中断可恢复，支持重试与（可选）断点续传。