交付标准(参考但不限于以下规则)

1. **封装形式**:

- 核心功能以 **Python Package** 形式交付 (如 doc_parser-1.0.0.tar.gz)，包含模块化代码、配置模板、示例脚本。
- 或提供 **HTTP 服务化部署选项**（基于 FastAPI/Flask)，支持通过 API 调用功能，适合跨语言集成。

2. 接口设计:

- 所有功能通过 **函数 / 类方法** 暴露，输入输出严格标准化（如输入为 dict/Path，输出为 dict/ 标准化文件路径)。
- 配置通过 **YAML/JSON 文件** 或环境变量传入，避免硬编码 (如 config.yaml 定义数据库连接、工具参数)。

3. 复用保障:

- 无业务相关依赖，仅依赖通用库（如 pandas/requests)，通过 requirements.txt 明确依赖版本。
- 提供详细文档：API 说明、配置示例、集成案例（如 “如何在 Django 项目中调用 OCR 模块”)。

模块 1：多源数据采集引擎

需求说明：构建通用数据采集工具，支持网页爬取、邮件监控、附件提取三大场景，解决不同来源非结构化数据的统一获取问题，输出标准化原始数据包。

- 针对企业 / 个人需要从多渠道（网页、邮件、本地文件）获取数据时，面临的 “采集逻辑分散、登录态管理复杂、附件分类混乱” 问题，开发一体化采集工具。
- 网页爬取需支持两类场景：无需登录的静态网站（通过 requests 获取 HTML)、需登录的动态网站（通过 Selenium/Playwright 模拟点击、输入账号密码，支持验证码手动输入)，并自动处理分页（如 “点击下一页” 直至结束)。
- 邮件监控需适配主流服务（如 Outlook/QQ 邮箱)：通过 IMAP 协议监听收件箱，按规则筛选 (如 “发件人 = xxx@company.com 且主题含‘报表’”)，自动下载附件并按类型分类（PDF/Excel/ 图片分别存入不同文件夹)。
- 配置需完全通过 YAML 文件定义：网页采集规则（URL、登录信息、需提取的元素如 “表格 / 列表”)、邮件筛选条件（发件人 / 主题 / 时间范围)、存储路径（本地文件夹或云端)。
- 异常处理需支持：网络中断自动重试（记录断点，恢复后从断点继续)、登录失败告

警（连续 3 次失败则通知用户）、损坏附件跳过（记录日志，不中断整体流程）。

- 核心目标是实现“多源数据一站式获取”，输出标准化数据包，为下游解析 / 分析提供原始素材。

技术要求：

- 编程语言：Python
- 核心库：
  - 网页爬取：Selenium/Playwright（复杂 JS 渲染网站）、requests.Session()（简单 Cookie 管理）
  - 邮件处理：imaplib（IMAP 协议）、microsoftgraph-python-sdk（Exchange/Outlook）
  - 附件处理：python-magic（文件类型识别）、os/shutil（附件保存）
- 功能：
  - 网页爬取：支持登录态保持、动态元素等待、多页面自动翻页
  - 邮件监控：按发件人 / 主题 / 时间筛选邮件，自动下载附件（PDF/Word/Excel）
  - 任务配置：通过 YAML 文件定义采集规则（如爬取 URL、邮件筛选条件）
  - 异常处理：网络中断重试、登录失败告警、文件损坏跳过

输出结果：

- 采集数据包（压缩包）：含网页 HTML / 截图、邮件正文（TXT/HTML）、分类存储的附件
- 采集日志：collection_log.jsonl（记录采集时间、来源、文件数量、错误信息）
- 配置文件示例：collection_config.yaml（可复用的规则模板）

考核要点：

- 多源兼容性：至少支持 2 类网页（需登录 / 无需登录）+1 类邮件服务的采集
- 数据完整性：附件下载成功率≥95%，网页关键内容（如列表、表格）无遗漏
- 可配置性：修改 YAML 即可切换采集目标，无需改动代码

加分项：

- 实现采集进度可视化（如进度条、剩余时间预估）
- 支持增量采集（仅获取上次采集后新增的数据）

扣分点：

- 硬编码采集目标（如固定 URL / 邮件地址，无法扩展）

- 无异常处理（网络错误直接崩溃，未保留已采集数据）

模块 2：第三方 API 统一集成引擎

需求说明：构建通用第三方 API 调用框架，整合电子签名、邮件通信、实体信件、搜索、存储 5 类主流 API，提供标准化调用接口与配置化管理，解决多平台 API 接口差异大、调用逻辑分散、密钥管理混乱的问题，支持快速切换服务商（如从 SendGrid 切换至 Mailgun）。

- 针对企业对接多类第三方服务（如邮件发送、电子签名、云存储）时，面临的 “API 接口差异大、调用逻辑分散、密钥管理混乱、故障无降级” 问题，开发标准化集成框架。
- 需覆盖 5 类核心 API：电子签名（DocuSign/PandaDoc）、邮件通信（SendGrid/Mailgun）、实体信件（Lob）、搜索（Google Search）、对象存储（AWS S3/MinIO），为每类 API 定义统一接口（如邮件服务的 send(to, subject, content)）。
- 配置化管理需简化接入：通过 YAML 文件指定服务商（如 “邮件服务 = SendGrid”）、密钥路径（如 Vault 中的密钥名）、默认参数（如发件人邮箱、S3 存储桶），切换服务商仅需修改配置。
- 容错机制需保障稳定性：API 调用超时自动重试（最多 3 次，间隔递增）、额度不足 / 服务故障时自动切换备用服务商（如 SendGrid 故障→切换至 Mailgun）、失败后记录详细日志（请求参数 / 响应 / 错误码）。
- 密钥安全需符合规范：敏感密钥（如 API Key）不落地存储（通过 Vault 获取），非敏感配置（如存储桶名）可写入 YAML，支持权限控制（仅管理员可修改配置）。
- 核心目标是让第三方 API 调用 “标准化、可切换、高可用”，适配企业办公、电商、内容平台等多场景。

技术要求：

- 编程语言：Python
- 核心库 / SDK：

- 电子签名：docusign-esign（DocuSign SDK）、pandadoc-api-client（PandaDoc SDK）
- 邮件通信：sendgrid（SendGrid SDK）、mailgun（Mailgun SDK）、microsoftgraph-python-sdk（Microsoft Graph 邮件模块）
- 实体信件：lob（Lob.com SDK）（可选）
- 搜索：google-api-python-client（Google Search API）
- 存储：boto3（AWS S3 SDK）、python-dotenv（环境变量管理）

- 功能：
  - 统一接口封装：为同类 API 定义通用方法（如 send_email()/create_signature_envelope()），屏蔽服务商接口差异
  - 配置化管理：通过 YAML 文件定义 API 服务商、密钥、默认参数（如邮件发件人、S3 存储桶）
  - 密钥安全：支持 HashiCorp Vault 对接（敏感密钥不落地存储）或本地加密文件存储
  - 异常处理：API 调用超时重试、额度不足告警、服务商切换降级（如 SendGrid 故障时自动切换至 Mailgun）
  - 调用日志：记录 API 请求参数、响应结果、耗时、错误信息，支持审计追溯

输出结果：

- API 集成工具包：third_party_api/（含封装后的类 / 函数、配置模板）
- 配置文件：api_config.yaml（定义服务商选择、密钥路径、默认参数）
- 调用示例：api_demo.py（含 5 类 API 的调用示例，如发送邮件、创建电子签名文档）
- 调用日志：api_call_logs.jsonl（结构化记录每一次 API 请求）

考核要点：

- 接口统一性：切换同类 API 服务商（如 DocuSign→PandaDoc）时，调用代码修改量≤2 行
- 多服务兼容性：至少完成 3 类 API（如邮件 + 存储 + 电子签名）的封装与调用验证
- 容错性：API 调用失败后重试成功率≥80%，支持手动触发服务商切换

加分项：

- 实现 API 调用量统计与成本预估（如按 SendGrid 邮件发送量计算月度费用）
- 支持 API 版本管理（适配不同服务商的 API 版本差异）

扣分点：

- 服务商接口硬编码（新增 / 切换服务商需大量修改封装逻辑）
- 密钥明文存储（如直接写在配置文件或代码中）
- 无异常处理（API 调用失败直接崩溃，无重试 / 降级机制）

思考：模块扩展功能：

- 通过本模块调用 DocuSign/PandaDoc 创建电子签名信封，完成签名流程；调用 AWS S3 SDK 将文档上传至对象存储
- 企业办公系统：集成电子签名（合同签署）、邮件通知（审批提醒）、云存储（文件备份）
- 电商平台：调用 Lob.com API 发送实体发票 / 会员信件，调用 AWS S3 存储商品图片
- 内容平台：调用 Google Search API 获取热点数据，辅助内容创作；调用邮件 API 发送订阅内容