

QuoteMind - 燕地鹏杨

→This application is dedicated to my mother's hardwork & personally driven to save manual management time on financial quotings to customers.

应用设想

需求	Solution
用户登录 <ul style="list-style-type: none">- 保存加密密码- 20分钟离开后自动退出登录	<ol style="list-style-type: none">1. Store only hashses of password2. Require re-login after 20mins of losing focus
能够设置客户信息: <ul style="list-style-type: none">- 公司- 联系人 (多人)- 地址- 客户 能够设置员工信息: <ul style="list-style-type: none">- 员工名字- 设计、安装角色 能够设置通用产品名称 - 通常涉及的材质 (例如选择车贴有防晒防水的材质) <ul style="list-style-type: none">- 公章 (蓝、红)- 地贴- 桌牌- 等等等 能够设置产品材质: <ul style="list-style-type: none">- 亚克力- 不锈钢- 拉丝- 写真布- 相纸 能够设置展品单位&规格 <ul style="list-style-type: none">- 张, 个, 寸、页、套、条、- Cm, mm, m	<ol style="list-style-type: none">1. Full stacking of copany information (creation, updates, deletion)2. Full stacking of employee details management3. Full stacking of management of material categories4. Full stacking of common product managements
添加交易记录 <ol style="list-style-type: none">1. 手动输入2. 自动输入 - 载入照片, 根据名称来自动录 取数据 (信息1+信息2+。。。) <ul style="list-style-type: none">• 自动打破分隔符号+, 留下可拖拽的信息• 公司 & 联系人• 产品 & 材质• 时间• 单价 (是否可以自动化)• 数量• 总价• 可附加多照片在 (同交易放在同一个文件夹)• 备注 -》同步截图信息名称保存记录• 未付款、已付款	

<p>交易汇总记录显示</p> <ul style="list-style-type: none"> • 按天、月、年显示交易 • 可细分到顾客 <p>**标明当前显示在多少订单未定价，多少未付款</p> <ul style="list-style-type: none"> • 筛选未定价，未付款 <p>**可pdf或excel形式导出报价提供给客户</p> <ul style="list-style-type: none"> • 日期 • 名称 • 数量 • 单价 • 合计 • 备注 	
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

使用需求

能够设置通用产品名称：

- 公章（蓝、红）
- 地贴
- 桌牌
- 证书
- 双色板
- 电子设计排版
- 宣传册
- 展板
- 异形
- 海报
- 车贴
- 电子表格
- 付印
- 扫描
- 洗照片
- 門牌
- 条幅

能够设置产品材质：

- 亚克力、透明亞克力
- 不锈钢
- 拉丝
- 写真布
- 相纸
- 啞膜
- 白膠
- 鋁合金
- 普通紙
- PVC
- KT

Technical Stack

MYSQL: A NoSQL database that can be used to store and retrieve data.

Flask: A web application framework that simplifies the process of building robust and scalable web applications.

React (Next.js): Next.js extends React by adding features like server-side rendering and automatic code splitting, enhancing the performance and SEO of the application.

Docker: A runtime environment for executing Flask on the server side. It allows for building scalable network applications

****Learn how to deloy your application****