Platform for Debt Collection Management

Overview:

The client is a specialist in offering debt collection solutions. Having operated in this field for 30 years, they sensed an opportunity to serve the market better by building a powerful, unified software platform that could perform a wide range of disparate functions. It translated to our having to develop an application that could streamline workflows, serve diverse user types, support integration with industry vendor APIs, and maintain powerful data security. Their demand was to ensure that the end product was flexible, secure and scalable. Also, depending upon the choice of the customer, it could be hosted either on cloud or on premise. Therefore, we built for them a multi-tenant platform, infused with the features that were desired, with the intent of driving innovation in serving the receivables industry.

Client Details:

Name: Confidential **Industry:** Finserv **Location:** USA

Technologies:

Python, React, Django REST, Docker, AWS, Kubernetes, Rancher, Linux, Ubuntu, GIT

Project Description:

This platform is designed mostly to target debt collection agencies with a focus on their needs and requirements. This platform can be used by multiple debt collection agencies according to their needs and procedures. It's a multi-tenant web application that holds each individual agency's data into different schemas with the same database.

The Multi-tenant web application has three basic end-users availability.

1. Staff
2. Client
3. Admin

Platform for Debt Collection Management

Staff: The staff of an agency can log in to this application and do any activities according to the permission level set by the agency. Agencies have the option to provide access to their staff by giving permission levels according to their activities.

Client: Clients are the agencies who provide loans in actuality with access to each data maintained by debt collection agencies.

Admin: Admin is the owner of the platform with access to all data and is able to access each client and staff's accounts to check entire data held by different handles.

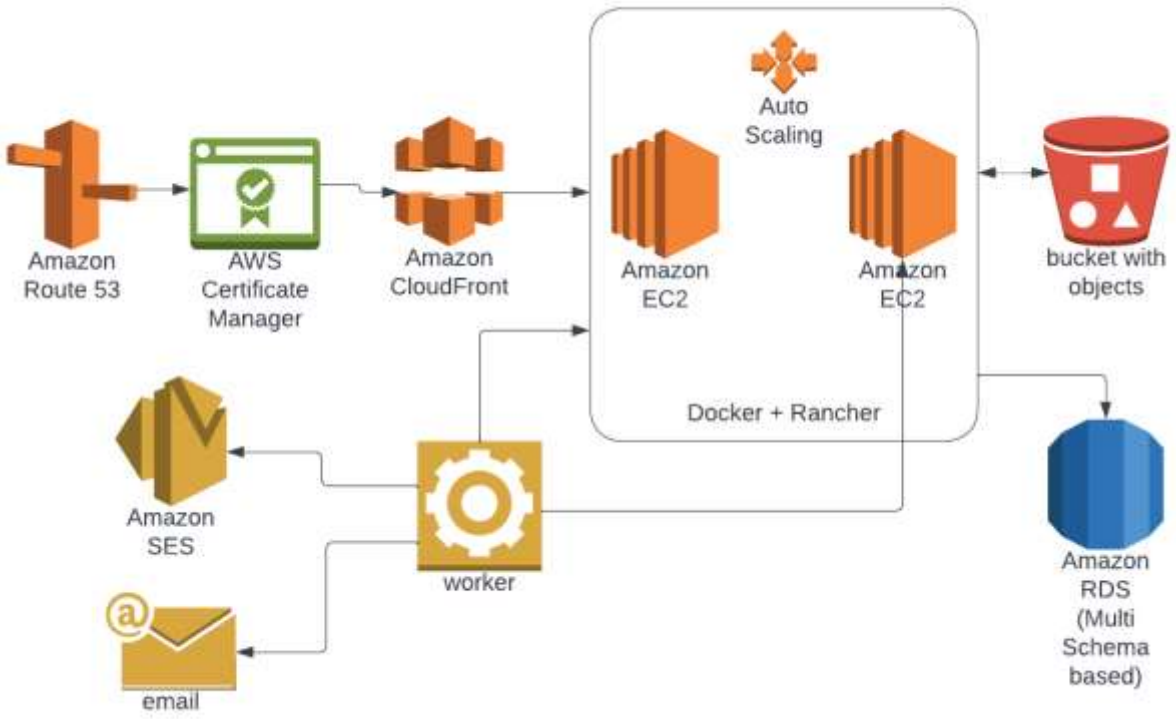
1. Staff can log in to the application and create accounts of debtors according to data transferred from the bank or any client with EDI.
2. They can create new Client records in the database.
3. They can create records of calls and payment plans according to debtors' convenient frequency and total amount.
4. Once payment is made, they can record the transaction for each debtor account.
5. If any debtor cannot be contacted, they can create legal matters and litigations according to their needs.
6. Since this is a multi-tenant application, users can find available codes by default, but further, they can create keys and codes according to their needs.

Methodology followed:

Development of the web platform is done with ReactJS, and the backend is powered by Python Django framework with the rest APIs. The database used is PostgreSQL.
AWS is used as a cloud platform for web apps as well as API deployment.
Rancher is used for container management with Kubernetes.
We followed Agile Scrum with Daily Standups and Sprint-based deliveries.

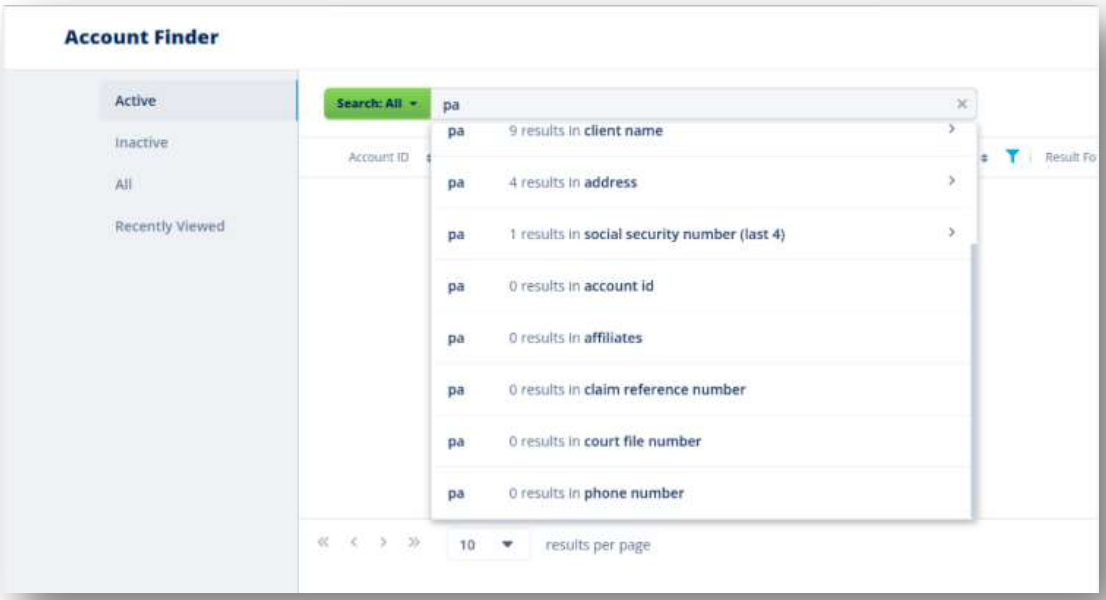
Platform for Debt Collection Management

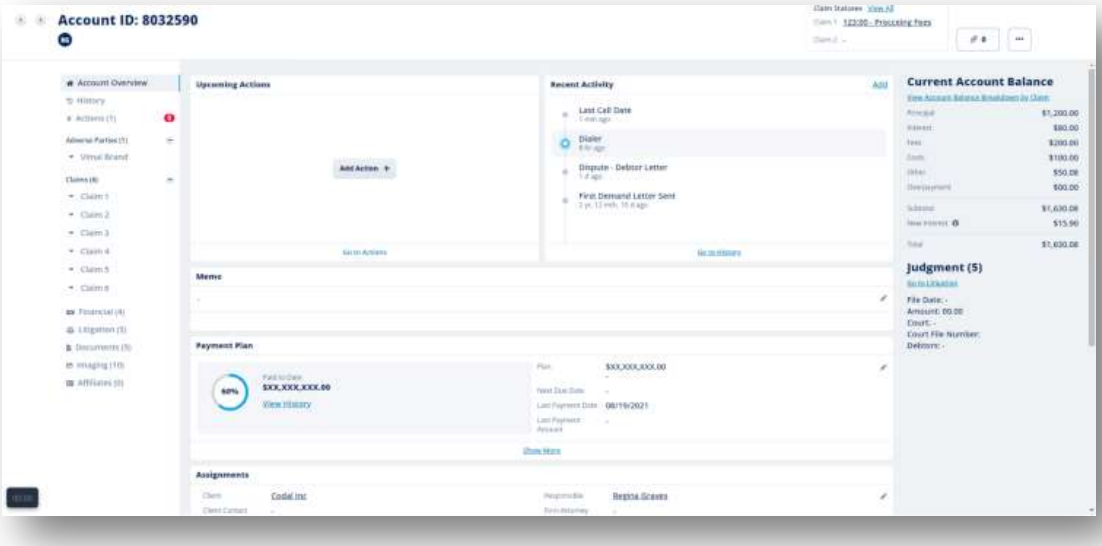
Architecture Design:



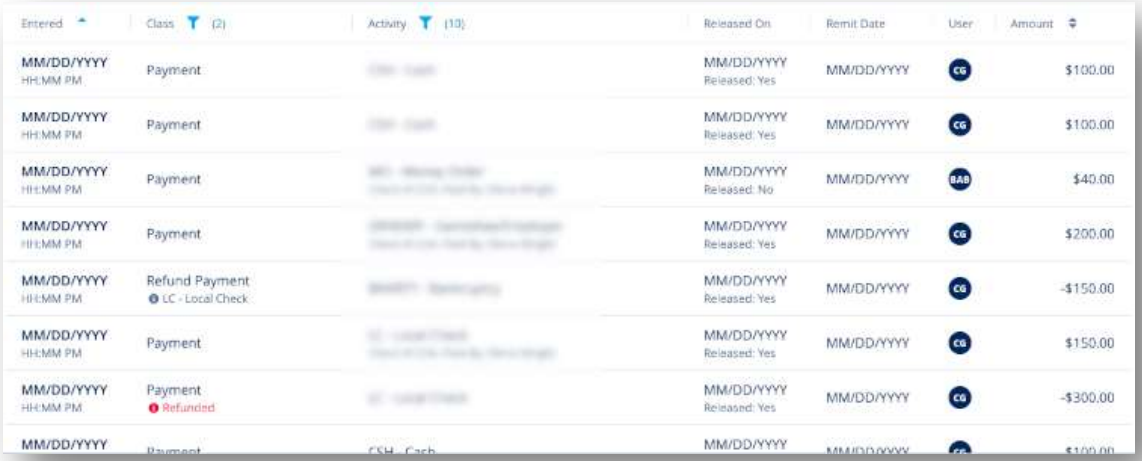
Platform for Debt Collection Management

Screenshots:





Platform for Debt Collection Management



Entered	Class (2)	Activity (10)	Released On	Remit Date	User	Amount
MM/DD/YYYY HH:MM PM	Payment		MM/DD/YYYY Released: Yes	MM/DD/YYYY	CG	$100.00
MM/DD/YYYY HH:MM PM	Payment		MM/DD/YYYY Released: Yes	MM/DD/YYYY	CG	$100.00
MM/DD/YYYY HH:MM PM	Payment		MM/DD/YYYY Released: No	MM/DD/YYYY	BAB	$40.00
MM/DD/YYYY HH:MM PM	Payment		MM/DD/YYYY Released: Yes	MM/DD/YYYY	CG	$200.00
MM/DD/YYYY HH:MM PM	Refund Payment LC - Local Check		MM/DD/YYYY Released: Yes	MM/DD/YYYY	CG	-$150.00
MM/DD/YYYY HH:MM PM	Payment		MM/DD/YYYY Released: Yes	MM/DD/YYYY	CG	$150.00
MM/DD/YYYY HH:MM PM	Payment Refunded		MM/DD/YYYY Released: Yes	MM/DD/YYYY	CG	-$300.00
MM/DD/YYYY	Payment	CSH - Cash	MM/DD/YYYY			$100.00