Contents

[Introduction & Overview 2](#_Toc40165831)

[Introduction 2](#_Toc40165832)

[Problem Statement 2](#_Toc40165833)

[Purpose of Solution 2](#_Toc40165834)

[Stakeholder Information 3](#_Toc40165835)

[Function Users 3](#_Toc40165836)

[High Level Design 4](#_Toc40165837)

[Subscription Portal 4](#_Toc40165838)

[Application Architecture 4](#_Toc40165839)

[Assumptions 5](#_Toc40165840)

[Out of Scope 5](#_Toc40165841)

[Technology Used 5](#_Toc40165842)

[Subscriber Services 6](#_Toc40165843)

[Application Architecture 6](#_Toc40165844)

[Assumptions 7](#_Toc40165845)

[Out of Scope 7](#_Toc40165846)

[Technology Used 7](#_Toc40165847)

[Resources 8](#_Toc40165848)

# Introduction & Overview

## Introduction

To provide interface for payment and account registration

## Problem Statement

There are thousands of debt collection agencies in US facing a challenge to digitally connect to their consumers and helping them with their various financial debt settlement related concerns, so to help all of these companies with their intended purpose we need to design an omnichannel communication platform which would primary have following objectives:

* Allow consumers to securely interact with Company via mobile app, web app, email, sms, chat bot and IVR
* Consumer would able to view Account Details, payment history, balance etc.
* Consumer or agent can securely able to exchange confidential documents with each other, consumer need to ability to digitally sign these documents so that they don’t have hassle to first print, sign and scan those documents
* Consumer can able to make payment to company through the digital payment gateways, system should have the ability to support multiple payment vendors as each company would have their own payment vendor.
* Consumer would have access to AI powered bot for self-help purpose and bot would be empowered to solve all the consumers problems within the ecosystem of solution

All the telephonic conversation between consumers and agents are already getting recorded in the recording software but system need to have ability to mine these recorded calls so that companies can able to understand are agent following all the regulations and compliances while talking to consumers.

## Purpose of Solution

To provide consumer able to subscribe to payment services to get and send payments online via multiple channels via mobile app, web app, sms, email, chat or IVR

To provide consumer register

To provide consumer ability to integrate payment channels via their on-site sites/apps to maintain documents/data on premise servers

# Stakeholder Information

## Function Users

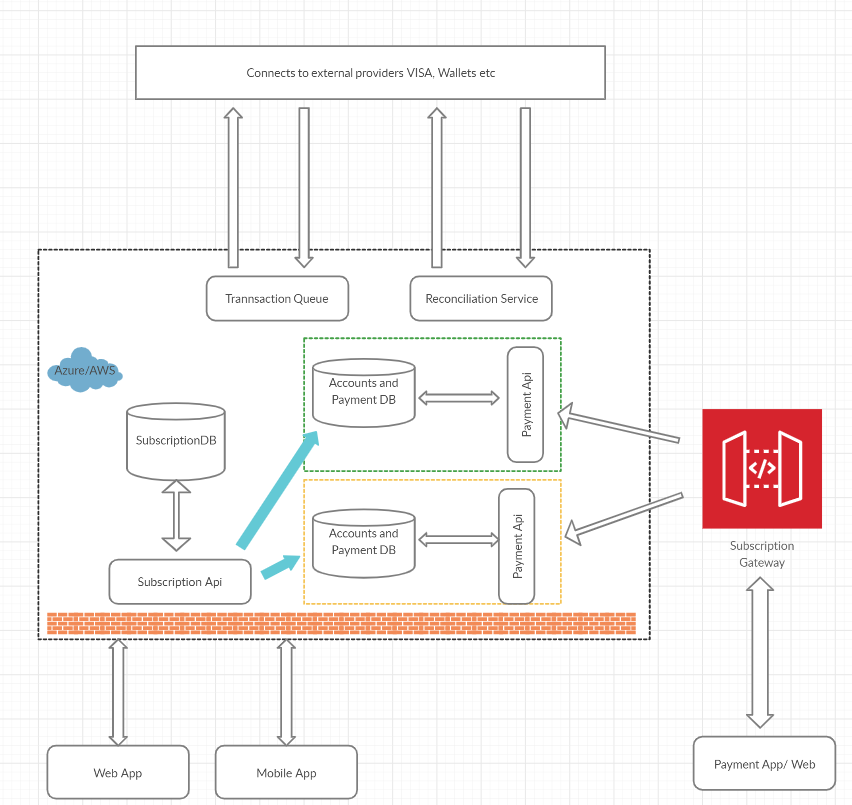
|  |  |  |
| --- | --- | --- |
| **#** | **Users** | **Description** |
| 1 | Consumer Admin Team | To Subscribe Payment service and modes via web, mobile, chat bots etc. |
| 2 | End User | To be able to register himself via Consumer web interface and send direct payments and schedule payments via multiple channels |
| 3 | Consumer Accounts Team | Interface to get account and payment history in form of reports |
| 4 | Consumer Vendors | Register consumer vendor account and payment interface |

# High Level Design

## Subscription Portal

Interface to for consumer admin where he can subscribe to services and get a “subscriptionkey” and “token” to connect

### Application Architecture



With every subscription new instance of partitioned DB will be created. “Payment Api” is partitioned based on “subscriptionkey”. With this subscription setup gateway will enabled for receiving subscriber’s request.

“Payment Api” will be responsible to send message/requests to Transaction Queue or Reconciliation Queue that interacts with banks/cards/bots.

### Assumptions

Minimum required details are saved in Payment DB to interact with bank/card

### Out of Scope

Refunds is currently out of scope

### Technology Used

Presentation Layer (Web App/ Mobile App/ Payment App/Web) – Responsive site in React/ Angular depending upon resources skillset

Database – SQL Server

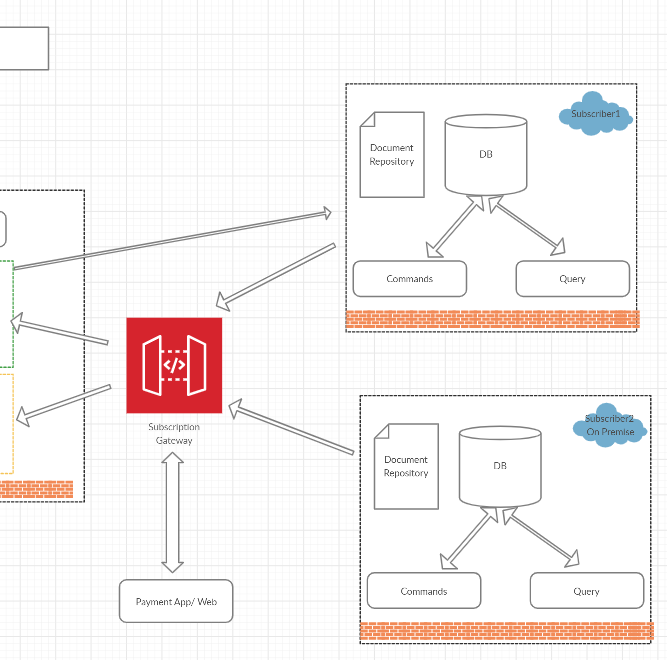
Payment API and Subscriber API - .Net Core 3.1 based on

Data Access Layer - Micro services pattern CQRS using RabbitMQ/AWS SQS/ Azure Queue

## Subscriber Services

System services to receive direct request from Consumer Portal or Payment Portal depending on subscription. Some customer want to use there portal with integration to our Api’s.

### Application Architecture



Identification of consumer subscription will be based on “subscription key” and “token”. Different endpoints will be provided to consumer for integration. Each command/endpoint can be called within from another command

1. **RegisterAccount**

*“Unique Identification”* will be shared that will be used for further transaction

1. **RegisterPayment**

*“Unique Identification”* will generated for each transaction between sender and payer. This will be used for further communication via network like payment status or EMI payment status

1. **GetPaymentStatus** – End point to get status of Registered Payment as some transactions take 2-3 days to get completed
2. **GetPayments** – To get summary of payment status with filtered parameter like fromData, toDate, status
3. **DataOutService** – If Subscriber has registered with this service than output/payment document like contracts can be send out to this “Path”/ “URL” depending upon setting

### Assumptions

Handling Data Sent out is handled via customer

### Out of Scope

Refunds is currently out of scope

### Technology Used

Presentation Layer (Payment App/Web) – Responsive site in React/ Angular depending upon resources skillset

Database – SQL Server

Payment Gateway Endpoints - .Net Core 3.1 based

Data Access Layer - Micro services pattern CQRS using RabbitMQ/AWS SQS/ Azure Queue

# Resources

1. Business Analyst (1)
2. UI/UX Designer (.5)
3. Angular/ React Developer (2)
4. .Net Core developer (5)
5. ML/AI Expert (.5)
6. Tester (3)
7. Deployment and Release (1)