

## **Overview**

Bill payment application user view consolidated bill payment status, can register new billers and make payments of bills using payment gateway or from wallets.

## **Functional Requirements**

Registration : Provide a feature to “register” customer for paying bills. As part of registration capture email id only. Successful registration should create a “stored value account” (wallet) associated with the registered customer. Option for customer to move funds to the wallet as part of registration.

Pay Biller : Pay your utility bills

- a) Fetch billers and bill from 3rd party external applications.
- b) Select a biller, fetch the bill and pay that using the wallet.
- c) Move funds from customer wallet to biller account.

Bulk/File based Bill Payment

- a) Process bill payments for N number of records received in a file. Multiple files can be received in a day. Design should support following use cases

## **Non Functional Requirements**

Logging & Monitoring

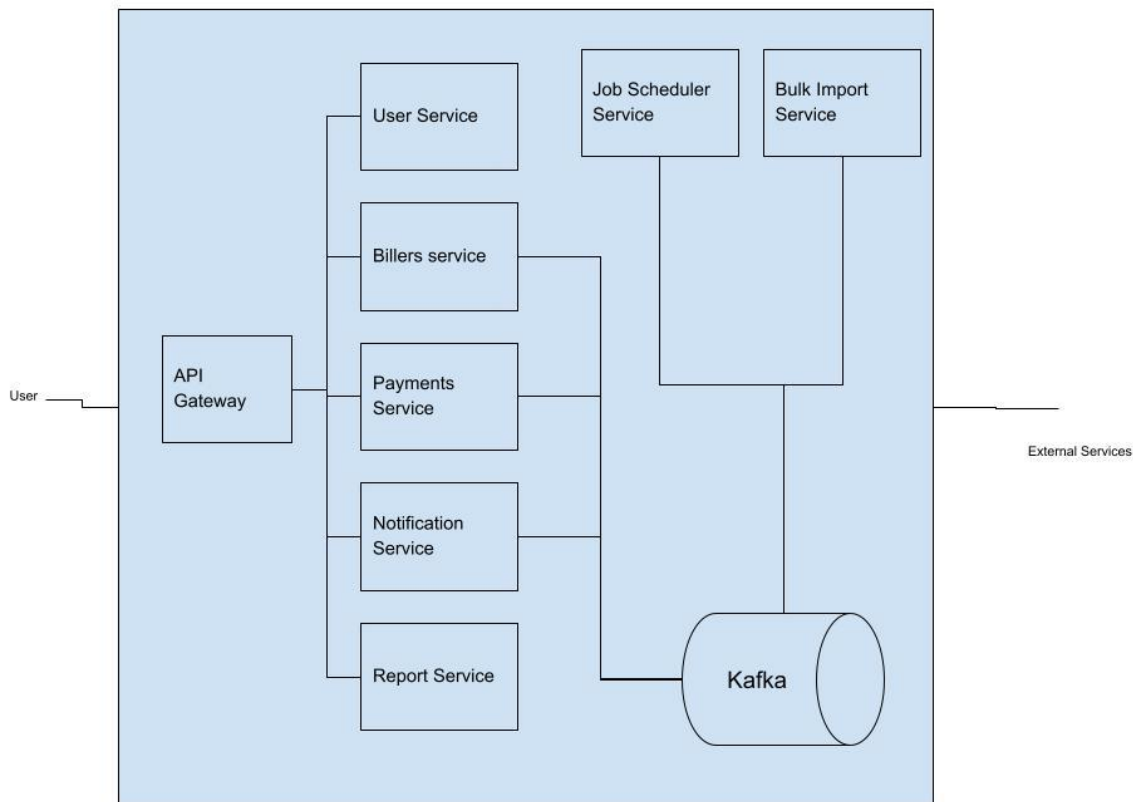
Auto scaling

CI CD processes and tech stack

Automation of test suites

Support PI/PCI Data. Application should be PCI complaint.

## Component Diagram



## Service Descriptions

### User service:

This service is responsible for managing user accounts, including storing and retrieving user information (e.g. name, address, payment method)

It also able to generate JWT token.

**Billers service:** This service is responsible for managing the user's bills, including storing and retrieving bill information (e.g. biller name, account number, amount, due date) and handling recurring payments and payment scheduling.

**Payment service:** This service is responsible for processing payment transactions and updating the user account in the user service accordingly.

Database: Mysql

**Notifications service:** This service is responsible for sending notifications to the user, such as bill reminders and payment confirmation messages.

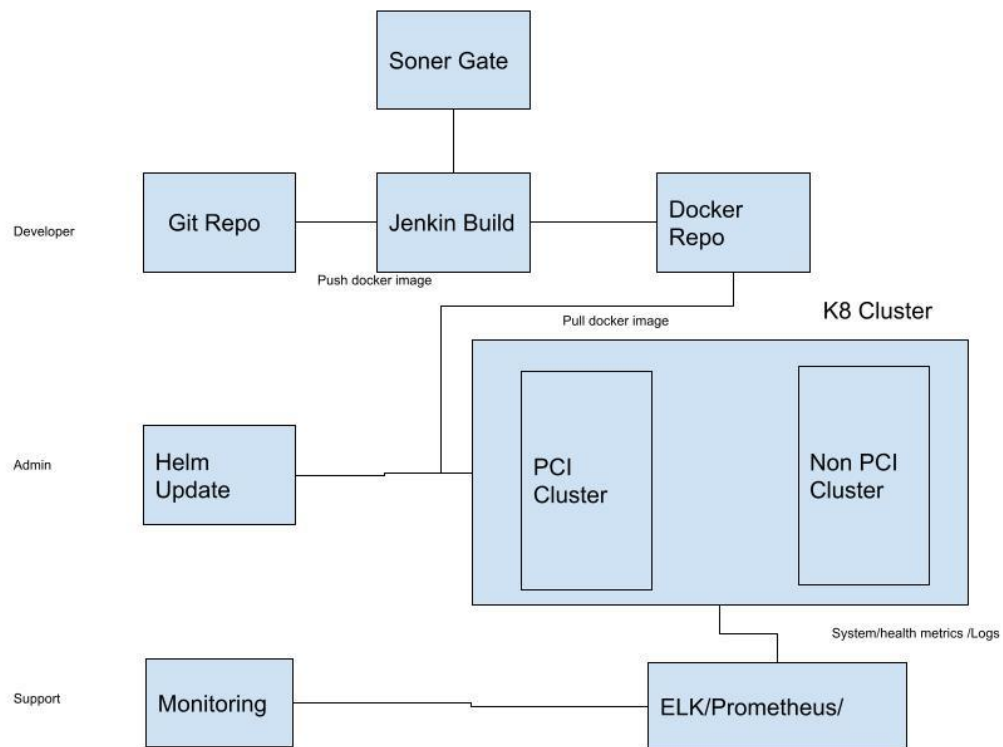
**Report service:** This service is responsible for storing CQRS data.

**Job Scheduler service:** This service is responsible for running various cron jobs like fetch bill amounts.

**Bulk Import service:** This service is responsible for Bulk payment processing from uploaded csv file.

**User interface (UI):** This service is responsible for providing the user with a way to interact with the system, including displaying the list of bills and payment history, and allowing the user to add and pay bills.

## Deployment Diagram



## Technology Stack

Java 1.8+  
Spring boot(spring security, spring cloud modules)  
Kuberneties  
Helm  
ReactJs/VueJs

Mysql  
Mongodb

Other Tools  
Git  
Jenkin  
ELK/Prometheus/Datadog

### **PCI compliance in container and Kubernetes environments**

1. secure network system firewall
2. not using any vendor default password
3. encrypt card holders data
4. protect system from vulnerability like malware by regularly updating antivirus, implementing strong access control
5. restrict physical access to the data.
6. Limit inbound Internet traffic
7. Secure cryptographic key distribution(trusted authority TA)
8. Monitor and Regularly test security systems and processes