

# Overview

The proposed solution (Cert@Chain) is a blockchain network, created by the Academic Institute (BEN). Institutes that are approved for issuance of certificate are registered in the network by the main Institute (BEN), and those institutes are permitted to add the certificates. These institutes can also delete the certificates in case of any modification needed. All Institutes, which have been approved can view the log of all certificates added in the Cert@Chain system. Also, the students can verify their certificate easily.

## Components

### I. Client

The client application is developed using Express & Node.js, with Handlebars templates for the web pages.

### II. Transaction Processors

The Cert@Chain system consists of only one Transactions Processor. The transactions processor is developed in Javascript.

# Users

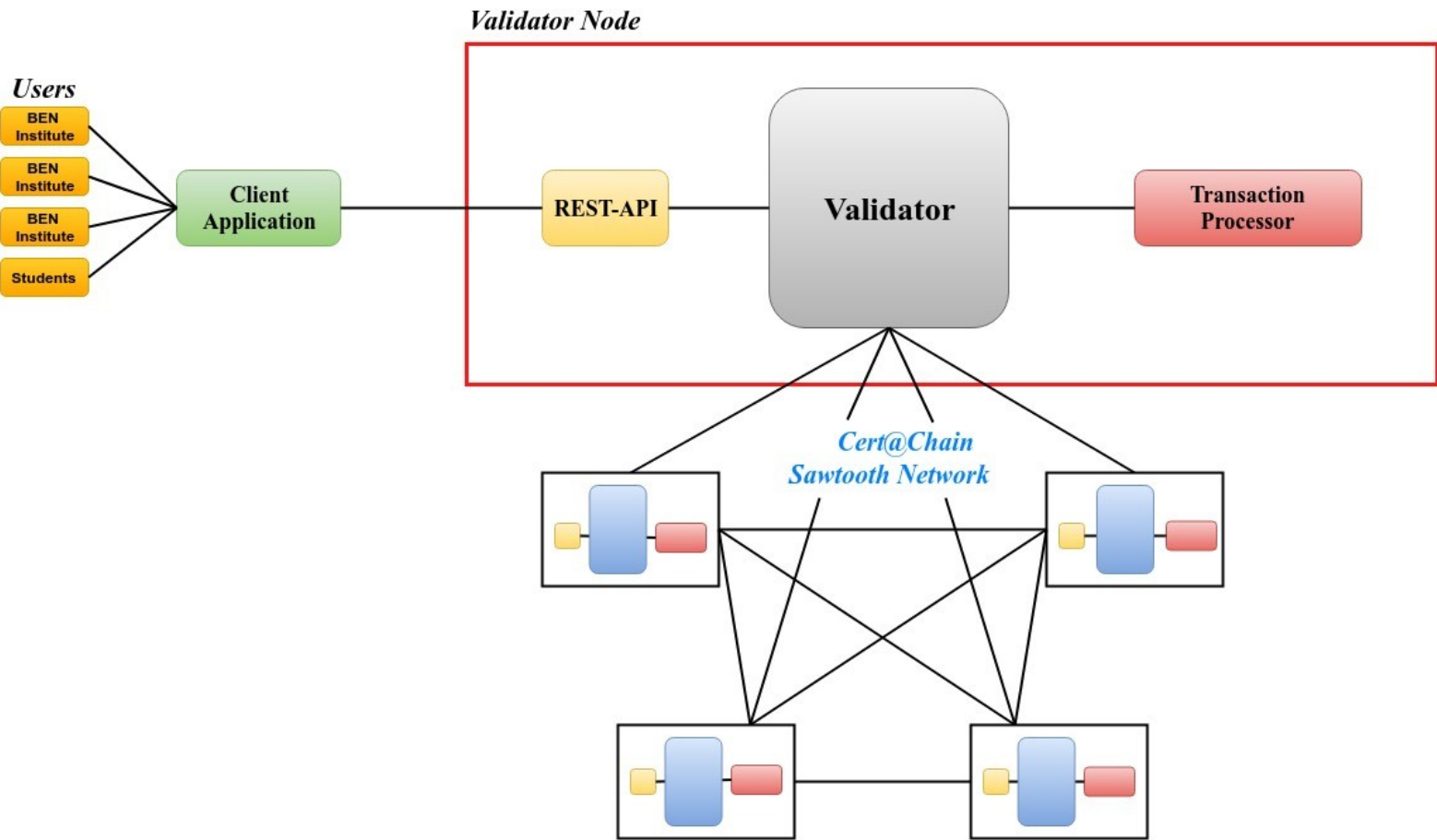
## I. BEN Institutes

The BEN Institutes, can view all the Certificates details in the Cert@Chain system. They can also drop the certificate incase of any modification needed.

## II. Students

The students can verify their certificate details using the valid USN.

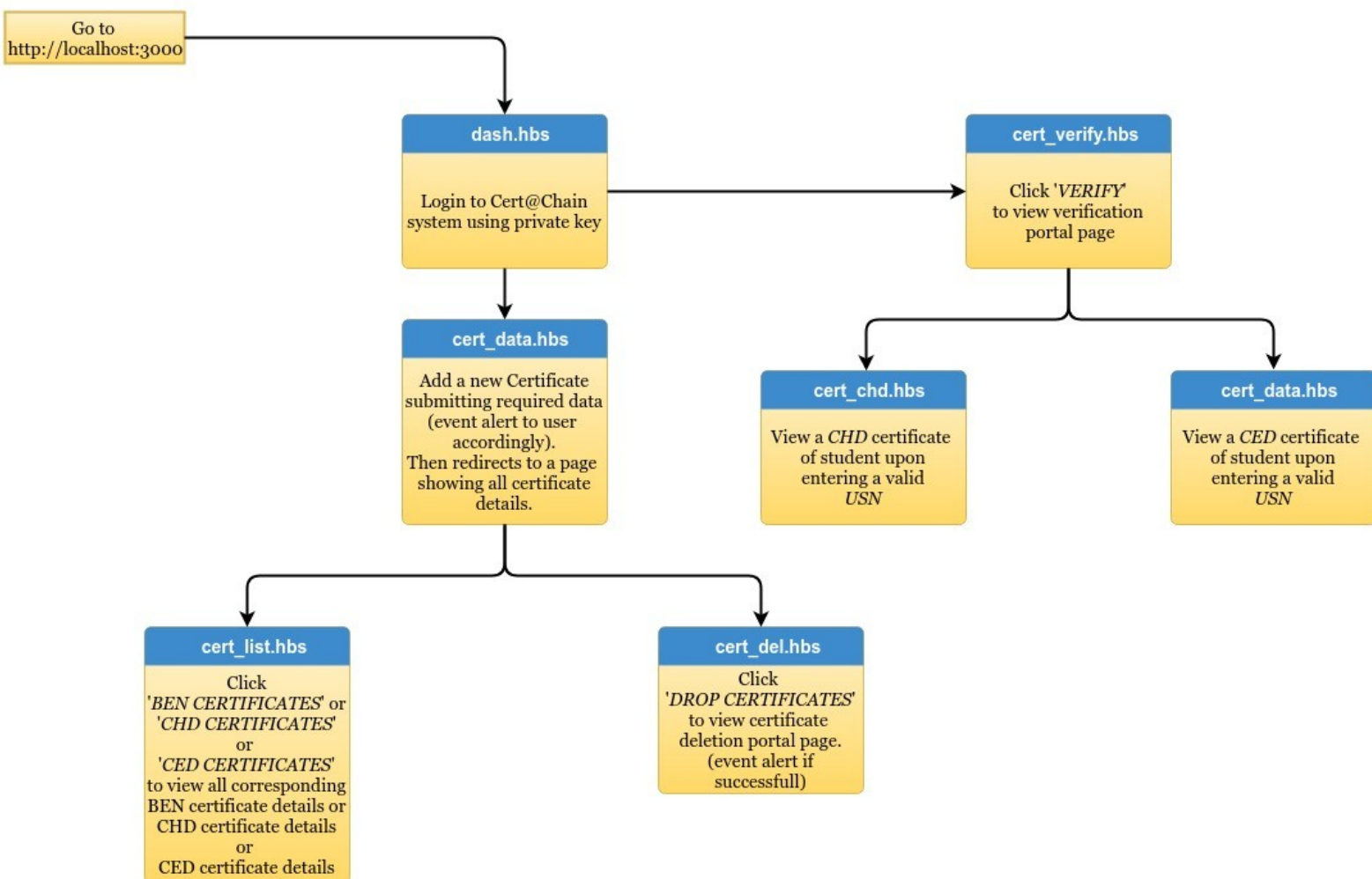
# Architecture Diagram



# Code Flow

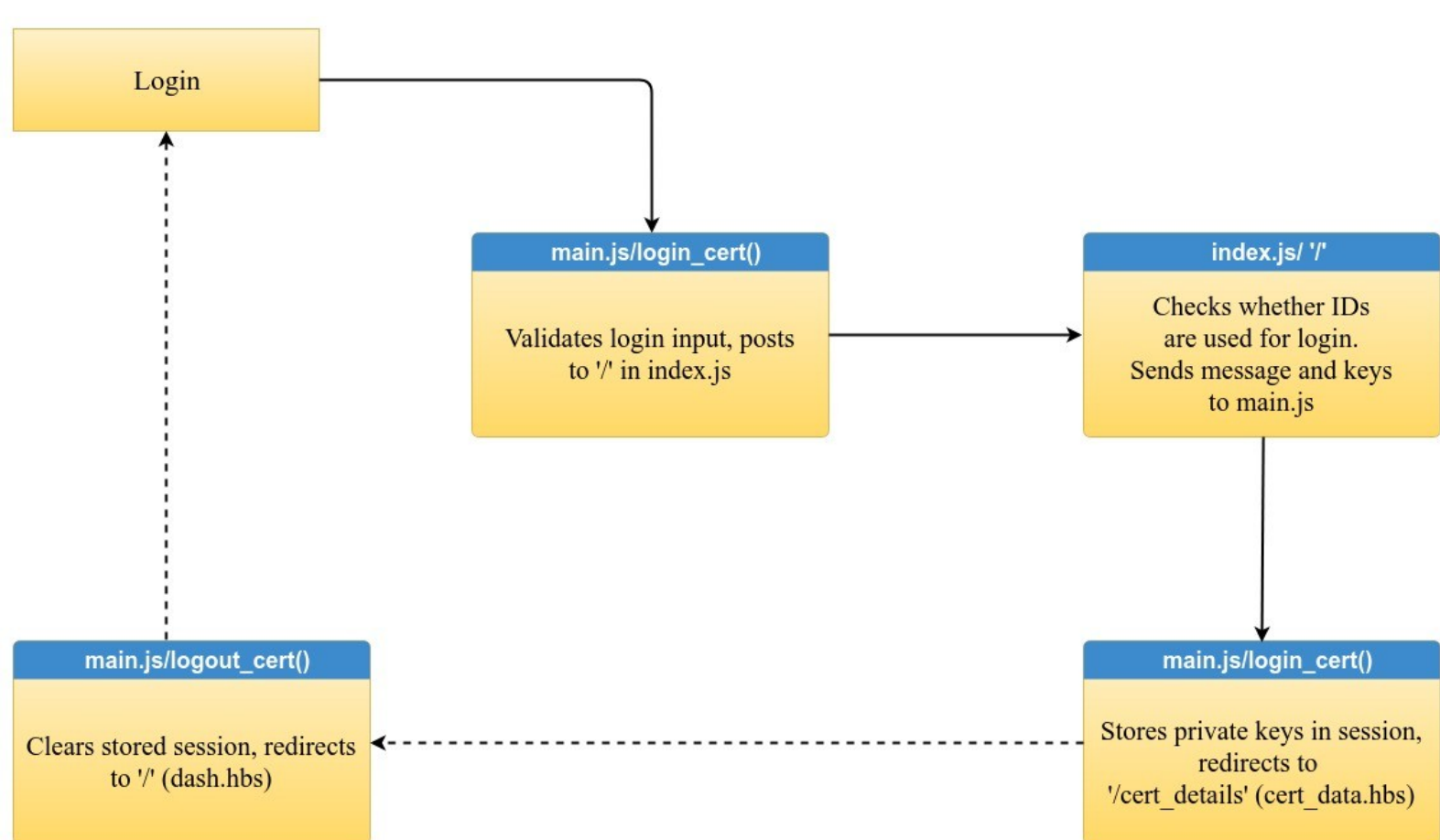
## I. User Interface

The below figure describes the code flow in the User Interface and the interactions between different components.



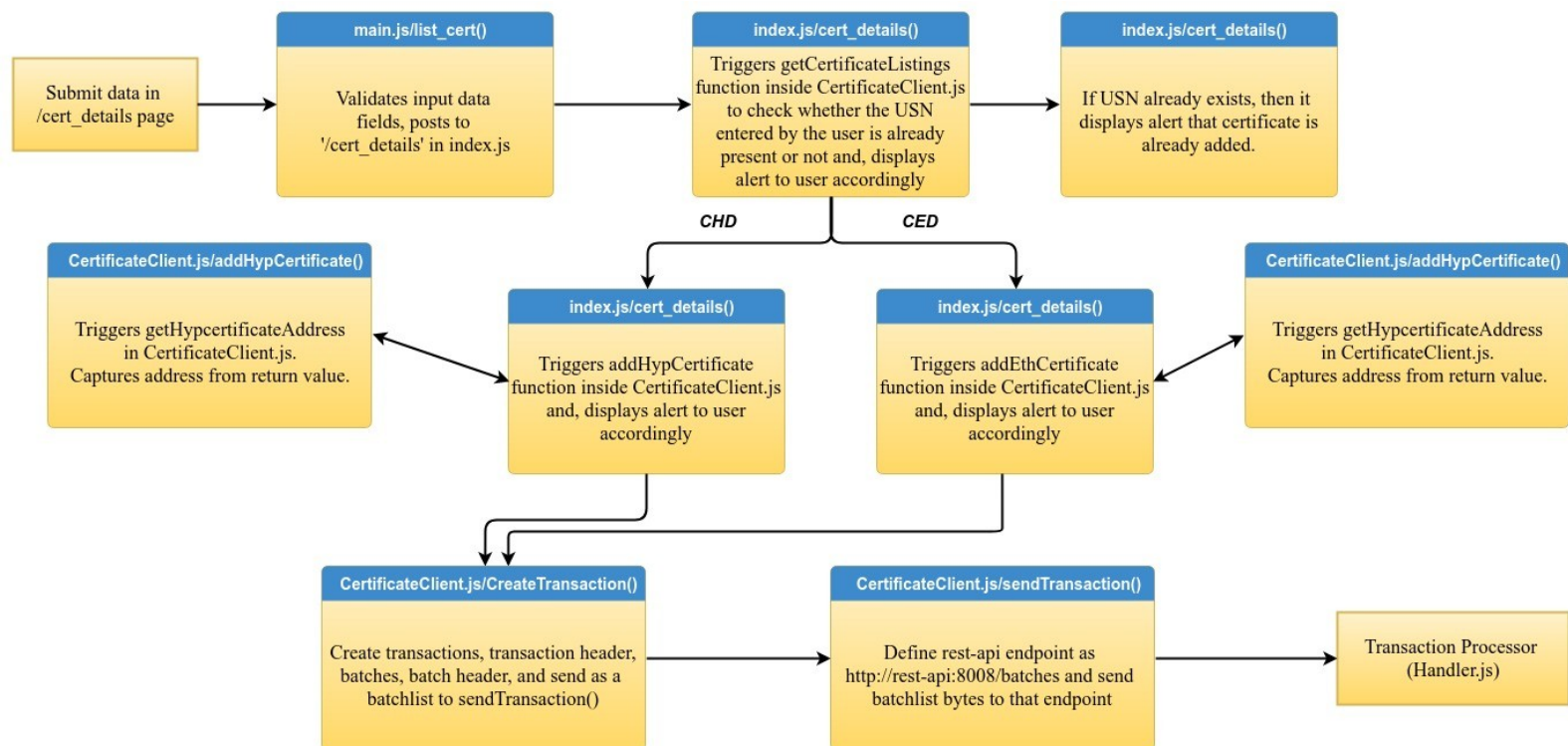
## II. User Login

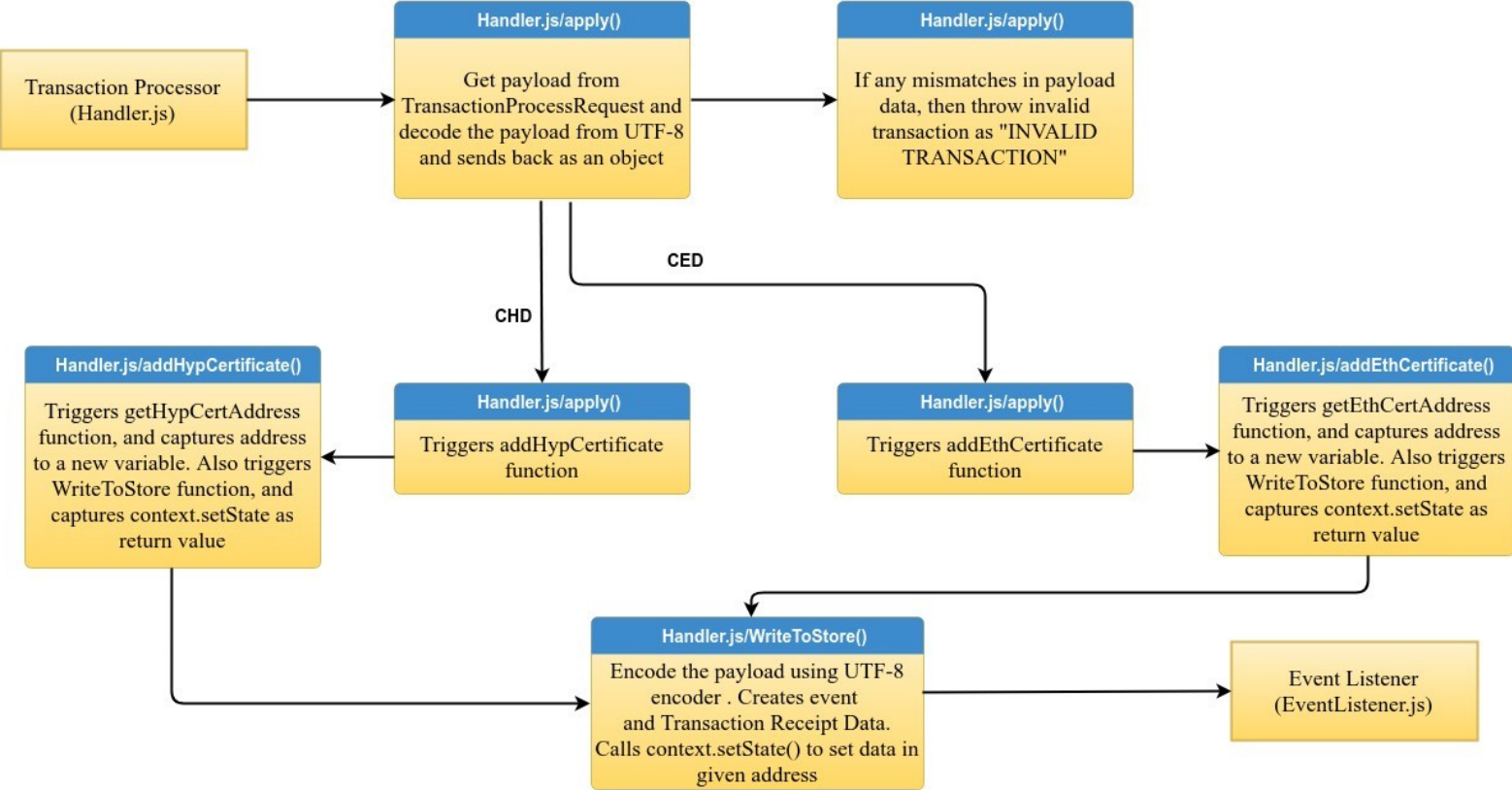
The below figure describes the code flow and actions involved when a user logs into the Cert@Chain system.



### III. Adding Certificates

The below figures describes the code flow and actions involved when a Certificate is being added by a BEN Institute into the Cert@Chain system.





## IV. Deleting Certificates

The below figures describes the code flow and actions involved when a Certificate is being dropped by a BEN Institute from the Cert@Chain system.

