Software Requirements Specification

For

**Blend Planning**

**in**

CSCTS PROJECT

**Prepared by Madhusudan Padmalochan**

**ELogic Square Analytics Pvt Limited**

**29-Nov-2020**

**Table of Contents**

[1.](#_1fob9te) Introduction 1

[1.1](#_3znysh7) Purpose 1

[1.2](#_tyjcwt) Document Conventions 1

[1.3](#_1t3h5sf) Intended Audience and Reading Suggestions 1

[1.4](#_2s8eyo1) Product Scope 1

[2.](#_3rdcrjn) Overall Description 2

[2.1](#_35nkun2) Product Perspective 2

[2.2](#_44sinio) Product Functions 2

[2.2.1](#_2jxsxqh) SECTION-1 DETAILS: CREATE BLEND PLAN 2

[a) Set Target output & Input parameters for resource constraints 2](#_z337ya)

[b) Enter LIFO feed: Add Rake/ Truck Tippler 2](#_3j2qqm3)

[c) Confirm usable stockpiles 2](#_1y810tw)

[d) Submit Blend Inputs to the Engine 2](#_4i7ojhp)

[2.2.2. SECTION-2 DETAILS: GENERATE BLEND PLAN 3](#_1ci93xb)

[a) Set the Run Mode 3](#_3whwml4)

[b) Set the Model Optimization Type 3](#_2bn6wsx)

[c) Generate & Compare various Blend Plans through System 3](#_3as4poj)

[d) Review the Selected Blend Plan (Create & Overwrite with a Manual Plan) and Approve 3](#_1pxezwc)

[2.2.3. SECTION-3 DETAILS: PLAN REPORTS 3](#_49x2ik5)

[2.3](#_2p2csry) User Classes and Characteristics 4

[2.3.1](#_147n2zr) Blend plan creation 4

[2.3.2](#_3o7alnk) Workflows 6

[2.4](#_ihv636) Operating Environment 7

[2.5](#_32hioqz) Design and Implementation Constraints 7

[2.6](#_1hmsyys) User Documentation 7

[2.7](#_41mghml) Assumptions and Dependencies 7

[2.7.1 . Proper Configuration of the Yard/ Stockpile Master 7](#_2grqrue)

[2.7.2. Proper Configuration of the Internal Vehicle Master 7](#_vx1227)

[2.7.3. One-Time configuration of the S-R usage matrix 7](#_3fwokq0)

[2.7.4 Blend Engine 7](#_1v1yuxt)

[3.](#_4f1mdlm) External Interface Requirements 8

[3.1](#_2u6wntf) User Interfaces 8

[3.2](#_3tbugp1) Hardware Interfaces 8

[3.3](#_nmf14n) Software Interfaces 8

[3.4](#_1mrcu09) Communications Interfaces 8

[4.](#_46r0co2) System Features 9

[4.1](#_2lwamvv) Description and Priority 9

[4.2](#_111kx3o) Stimulus/Response Sequences 9

[4.3](#_3l18frh) Functional Requirements 9

[5.](#_4k668n3) Other Nonfunctional Requirements 9

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Madhusudan Padmalochan | 29-Nov-2020 | Initial Draft | 0.1 |
| Madhusudan Padmalochan | 18-Dec-2020 | Workflows Updated | 0.2 |

# Introduction

## Purpose

*The purpose of the document is to define the details of the CSCTS (Coal Supply Chain Tracking System) to all the stake holders on the process of the coal movement, management in the plants*

## Document Conventions AKP Suggest : keep this in order of alphabet

|  |  |
| --- | --- |
| *API* | *Application Programming Interface* |
| *CHP* | *Combined heat and power* |
| *CSCTS* | *Coal Supply Chain Management System* |
| *DAO* | *Data Access Object* |
| *DB* | *Database* |
| *Dozer* | *Dozer* |
| *HHD* | *Handheld Device* |
| *Hyva* | *Internal vehicles of the Plant* |
| *LIFO* | *Last In First Out* |
| *Rake* | *Railway Rake* |
| *Source* | *Source of the Coal* |
| *SR* | *Stacker Reclaimer* |
| *Supplier* | *Coal Mines* |
| *Transporter* | *Transporter of the Coal* |
| *Truck* | *Vehicle carrying the Coal* |
| *UI* | *User Interface – Web Pages* |

## Intended Audience and Reading Suggestions

*The document is intended to all the stake holders of the product like developers, project managers, delivery partners, testers and the plant teams and the CHP Team.*

*The document should be read in the above defined format so that all the flow is observed as defined.*

## Product Scope

*The purpose of the CSCTS is to provide the details of the coal movement inside the plant, starting from the in bound to storage to consumption. It is also targeted at providing the movement of the truck through near real-time view of the status of the trucks. The anomalies are detected and directed to the concerned stake holders for further actions.*

# Overall Description

The blending module is used to generate a blend plan against a set of target output parameters and input parameters (stockpile related) specified by the user. The user can go ahead with the execution of the system generated plan or modify the plan to suit their needs.

## Product Perspective

The CSCTS system stores the following the following information

* Target output parameters
* Stockpile specific input parameters
* Details of the plan generated (system generated and manual-if any)

## Product Functions

### 2.2.1 SECTION-1 DETAILS: CREATE BLEND PLAN

This section allows User to enter target output values and input parameters for a Blend Plan.

The user input to be given is:

#### a) Set Target output & Input parameters for resource constraints

User needs to enter values against the UI fields under the Create Blend Plan Screen of the Blending Module.

There are two divisions: target parameters and input parameters. The target parameters are the output required by the user and the input parameters deal with the stockpile parameters.

#### b) Enter LIFO feed: Add Rake/ Truck Tippler

If the user wants to enter rake or tippler data, then they shall enter values against the UI fields by clicking on the “Add Truck Tippler Data” and the “Add Rake Data” buttons in the Create Blend Plan Screen.

#### c) Confirm usable stockpiles

User needs to click on “Get Stockpile Details” button which will fetch the list of saved stockpiles.

The following conditions must hold good for a stockpile to be displayed:

1. Stockpile is Active
2. Active Outward flag is Active
3. At Least 1 reclamation mode (S-R, Hyva, Dozer) is set as True

#### d) Submit Blend Inputs to the Engine

This is the last step for the user on this section on “Create Blend Plan”. User, after verifying all the input values across (a), (b), and (c) above should click on the “Submit” button to freeze the input values and push it to the Blend Engine for processing.

A message is displayed “Blend Input Captured”. User is redirected to the 2nd Section “Generate Blend Plan”.

# 

### 2.2.2. SECTION-2 DETAILS: GENERATE BLEND PLAN

This section allows User to query the Blending Python Engine for fetching recommended Blend Plan details against various run modes.

The steps for using this section is as specified below:

#### a) Set the Run Mode

Users need to select the run mode for the Blend Model: normal or heap constraint.

#### b) Set the Model Optimization Type

Users need to specify the model type for optimization consideration by the Blend Model. The 5 types of optimization models: cost saver, energy saver, LIFO, age priority and hybrid. AKP Query : difference between LIFO and Age priority

|  |  |
| --- | --- |
| **Optimization Type** | **Description/ Details** |
| **COST SAVER** | The model will minimize total recipe cost for Blend Output. |
| **ENERGY SAVER** | The model will prioritize the stockpiles from which maximum energy loss is expected as on date (by using the defined GCV Loss value Kcal/ Kg/ Month from stockpile master data). Thus, the output will try to minimize the energy loss in the Yard. |
| **LIFO** | The model gives highest priority to LIFO (Rake and Truck tippler) Feed (entered in section-2.2.1 a) and then minimize the recipe cost for the complementary blend required for achieving target output. |
| **AGE PRIORITY** | The model will prioritize stockpiles approaching closer to the defined age limit. |
| **Hybrid** | A combination of AGE and COST type. The weightage of each type can be specified in section 2.2.1 a |

#### c) Generate & Compare various Blend Plans through System

By repeatedly selecting combinations of Run Mode & Optimization Type (from (a) and (b) above), the user can generate various Blend Plans and compare them on the UI.

#### d) Review the Selected Blend Plan (Create & Overwrite with a Manual Plan) and Approve

The System Blend Plan selected by the user in the previous step can be modified (by either changing the quantity of the selected stockpiles or by adding/ deleting new stockpiles in the blend plan)

The Blend Plan is successfully recorded in the system and a suitable message is displayed to the User. Users also get automatically redirected to the “Create Blend Plan” home-screen.

### 2.2.3. SECTION-3 DETAILS: PLAN REPORTS AKP Query : Approve and execute through Report?

This section contains the historical Blend Plans created and saved by users.

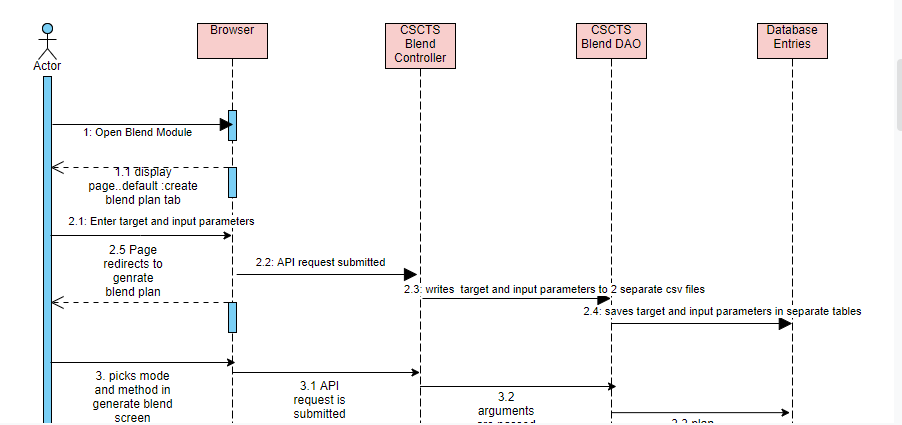
The users can get the list of approved blend plans created between the specified “From” and “To” Date entered by the user on this UI. The Reports Section also has the provision for authorized users to quality/commit a blend plan for Execution. Such plans will now start appearing in the Job-Execution Use-Case for the users to Execute

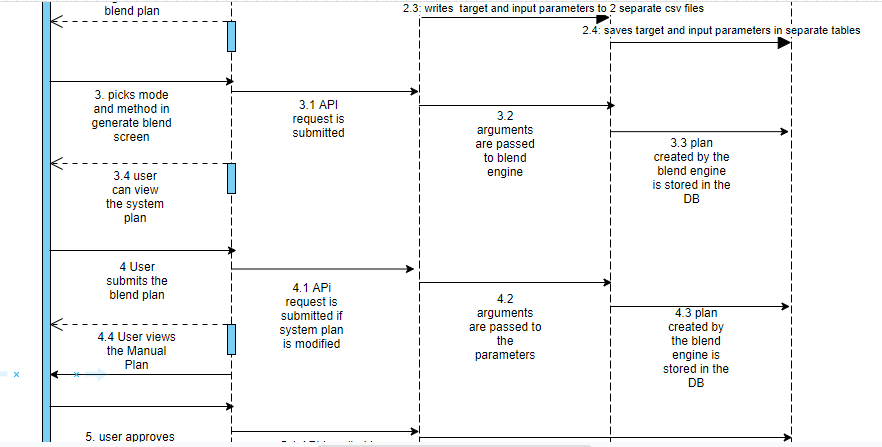
AKP Query : Validation need to be explained. Once executed to the quantity, will it be automatically deactivated / executed flagged?

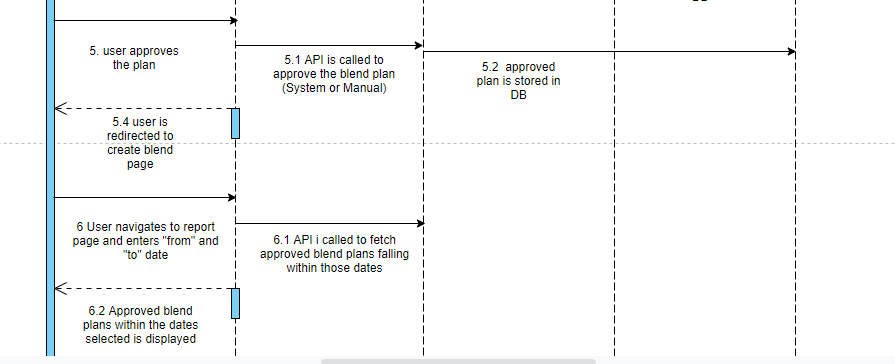
The details of the Job-Execution are detailed out in the Job-Allocation SRS. In a nutshell, the job execution is in auto-mode once the User selects a plan to be executed. The plan status is also available and updated periodically while the job is in execution mode

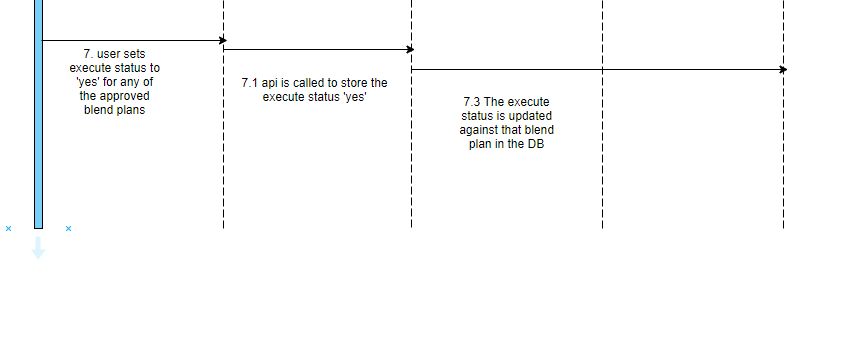
## User Classes and Characteristics

### Blend plan creation



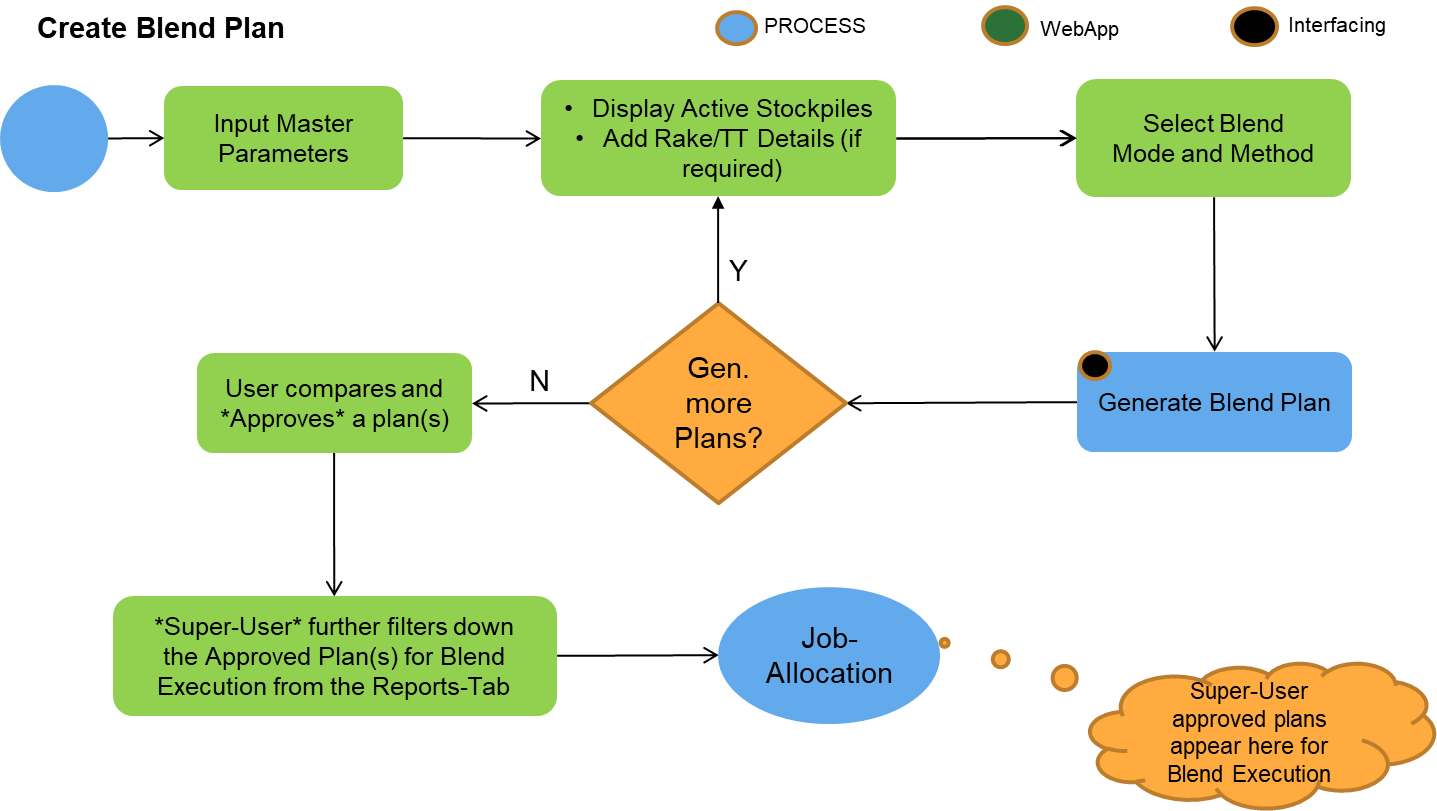






### Workflows AKP Observation : Workflow is confusing. So needs clarification

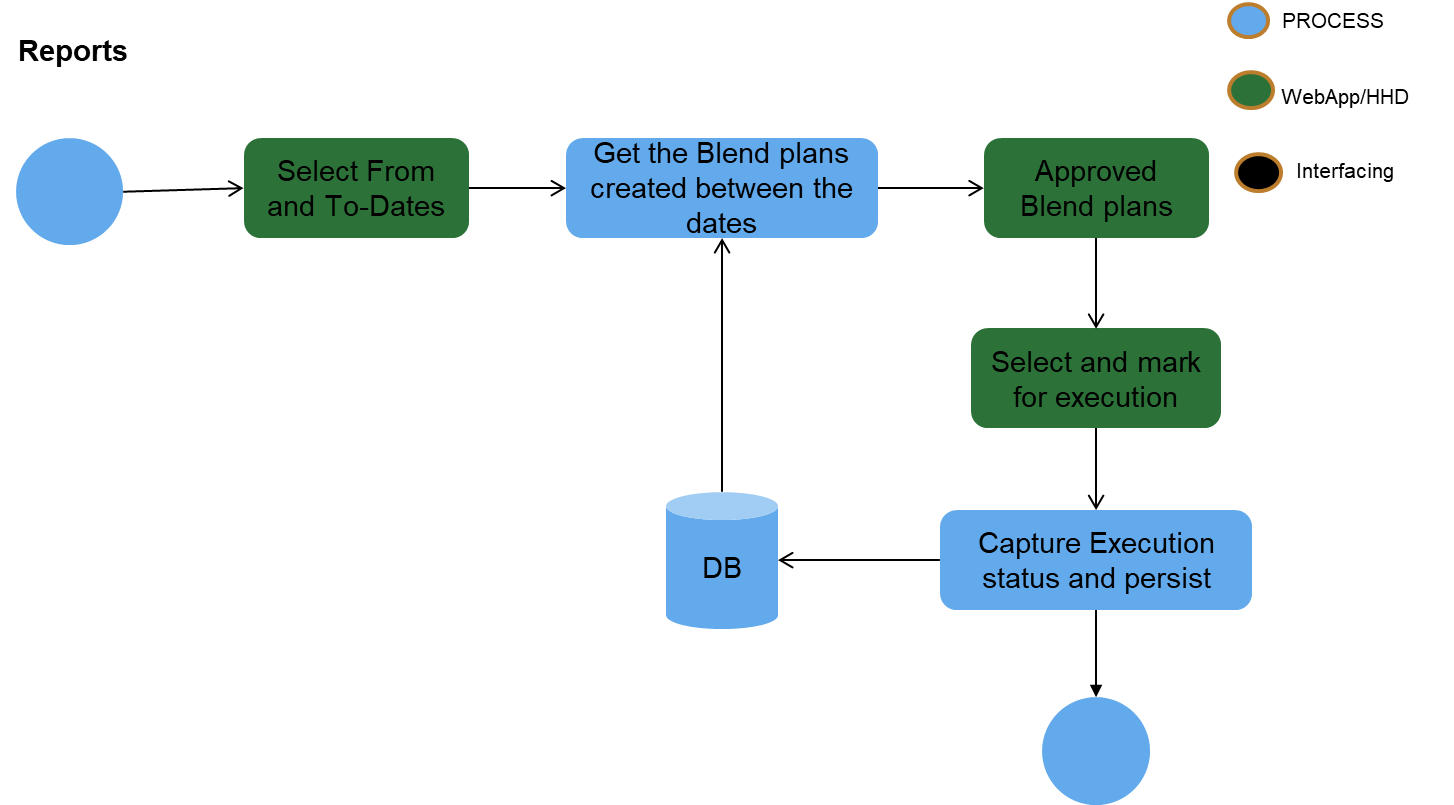
Workflow has been edited and simplified



1. User would key-in / adjust the master parameters according to the Blend Plan
2. User would select the active stock piles and also add Rake and TT details (optionally)
3. User further selects the Blend Mode and Method, and generates the ‘Blend Plan’
4. User could iterate further and create more plans based on the ‘Mode and Method’ of their choice
5. User would compare, select and approve the Blend Plan’ most appropriate
6. Super-User, would further skim through the ‘approved plans’ and select for ‘Blend Execution’
7. Such plans would appear in the ‘Job-Allocation’ screen for the user to select for ‘Blend Execution’

AKP Query: Where is it approved in Report workflow?

The user has two options: “Record Updated Plan” (Manual Plan) or “Continue System Plan and Exit” (System Generated Plan) in the ‘Generate Blend Plan’ section. On clicking on any one of these options, the plan is approved by the system and appears in the Report. Where, further user can select one of the approved plans for ‘Blend-Execution’



## Operating Environment

Operating environment for the CSCTS application is as below

* Oracle database
* Operating System: Centos Linux
* Client: Browser
* Platform: Java, Apache Ignite, Angular 8

## Design and Implementation Constraints

* Proper Configuration of the Yard/ Stockpile Master
* Proper Configuration of the Internal Vehicle Master
* One-Time configuration of the S-R usage matrix

## User Documentation

Module wise user manual is provided during the feature releases.

## Assumptions and Dependencies

### 2.7.1 . Proper Configuration of the Yard/ Stockpile Master

This module can be accessed under the “Yard & Stockpile Master” section in the “CHP” of the CSCTS WebUI menu.

### 2.7.2. Proper Configuration of the Internal Vehicle Master

This module can be accessed under the “Internal Vehicle” section in the “Asset Master” of the CSCTS WebUI menu.

### 2.7.3. One-Time configuration of the S-R usage matrix

This is a compatibility matrix which is configured at the back-end for mapping the parallel operation compatibility of the heaps when reclaimed through S-R. It specifies that during an S-R run job, which 2 heap can be simultaneously operated.

### 2.7.4 Blend Engine

The blend engine is an optimization algorithm. Python version 3.6 or higher is required to run this.

# External Interface Requirements

## User Interfaces

Front End Interface: Angular

Middle End Interface: Java Rest API’s

Backend Interface: Oracle

Standards for User Interface:

## Hardware Interfaces

Linux – Centos 7.0

A browser which supports HTML and Java Script

## Software Interfaces

Following are the software used for the CSCTS application

|  |  |  |
| --- | --- | --- |
| **Software Used** | **Version** | **Description** |
| Java | Java 1.8.0\_u231 | To build the middle layer of the application, we have used Java |
| Apache Ignite | 2.7.5 | Ignite is used as an in-memory cache layer for the frequently used data |
| Oracle | 12.c | To save all the data related to the coal management |
| Angular | 8 | To create the user interfaces |
| Linux | Centos 7.0 |  |
| SMTP | In –house | Email Integration |
| SMS | SMS Gateway | SMS Integration |
| Python | Python 3.6 | To run the blend engine |

AKP Query : Python 3.6 for execution is required. Should be a part of this table 3.3.

## Communications Interfaces

* Use a web browser to access and manage the blending module

# System Features

### 4.1 Description and Priority

The Blending screen shall provide the user the option to generate the blend plan, modify and approve it.

### 4.2 Stimulus/Response Sequences

* *create blend plan*
* *generate blend plan*
* *recalculate output*
* *approve blend plan*
* *reports*

### 4.3 Functional Requirements

* The create blend plan screen shall allow the user to enter the target and input parameters
* The generate blend plan screen shall enable the user to generate a blend plan based on certain mode and methods.
* The user shall have the option to modify the system generated blend plan parameters and generate a new plan.
* The reports section shall allow the user to view all the approved blend plans within the dates selected and enable it for execution.

# Other Nonfunctional Requirements

* CSCTS modules or pages developed should be supported by Chrome and Edge
* CSCTS Web average page response should not be more than 5 secs
* Any or all CSCTS Web or HHD modules / functions should be accessed only by valid logged credentials
* Any or all operations performed should be audited / logged in CSCTS
* Any or all CSCTS Web pages will follow or adhere to these User Guidelines Principle