**Alixpartners**

**GTO: ICE**

**Technical Design Document**

**Version 0.1**

**Revision History**

| **Sr. No.** | **Date of Revision** | **Version** | **Description Of Change** | **Prepared / Updated By / Date** | **Reviewed By / Date** | **Approved By / Date** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 15-Nov-2022 | 0.1 | Technical design of the ICE application | Vikrant Kulkarni | NA | NA |
|  |  |  |  |  |  |  |

**Template Revision History**

| **Sr. No.** | **Date of Revision** | **Version** | **Description Of Change** | **Prepared / Updated By / Date** | **Reviewed By / Date** | **Approved By / Date** |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

DISCLAIMER

**This document is strictly private, confidential and personal to its recipients and should not be copied, distributed or reproduced in whole or in part, nor passed to any third party.**

[1. Introduction](#_heading=h.3znysh7) **5**

[1.1. Background / Business Problem / Business Overview](#_heading=h.2et92p0) **5**

[1.2. Purpose](#_heading=h.tyjcwt) **5**

[1.3. Related Documents](#_heading=h.3dy6vkm) **5**

[2. Scope of Work](#_heading=h.4d34og8) **6**

[2.1. Front End: Visualizations / UI / Dashboard](#_heading=h.2s8eyo1) **6**

[2.2. Back End: Data Integration / Storage](#_heading=h.17dp8vu) **7**

[2.3. Out of Scope](#_heading=h.3rdcrjn) **7**

[2.4. Dependencies/Constraints/Risk/Gaps](#_heading=h.26in1rg) **7**

[3. Definitions](#_heading=h.lnxbz9) **7**

[4. Architecture Overview](#_heading=h.1ksv4uv) **8**

[4.1. Understanding of Current System Architecture](#_heading=h.44sinio) **8**

[4.2. Architecture](#_heading=h.2jxsxqh) **8**

[**4.2.1. System Characteristics**](#_heading=h.z337ya) **8**

[**4.2.2. Process Flow**](#_heading=h.3j2qqm3) **8**

[**4.2.3. Environment/System Configuration**](#_heading=h.1y810tw) **9**

[**4.2.4. Technology Stack**](#_heading=h.4i7ojhp) **10**

[**4.2.5. Data Model**](#_heading=h.2xcytpi) **10**

[**4.2.6. Detailed data model**](#_heading=h.1ci93xb) **10**

[Table Name: Dashboard\_tableau](#_heading=h.2bn6wsx) 11

[Table Name: country\_mapping](#_heading=h.qsh70q) 13

[Table Name: dim\_project](#_heading=h.3as4poj) 15

[Table Name: dim\_cost\_estimation\_commodities](#_heading=h.1pxezwc) 17

[Table Name: dim\_project\_cost\_details](#_heading=h.49x2ik5) 19

[Table Name: target.cost\_estimation\_summary](#_heading=h.2p2csry) 21

[Table Name: target.dim\_spec\_material](#_heading=h.147n2zr) 22

[Table Name: target.dim\_subsection](#_heading=h.3o7alnk) 24

[Table Name: target.dim\_conversion\_library](#_heading=h.23ckvvd) 25

[Table Name: target.baseline\_costs](#_heading=h.ihv636) 27

[Table Name: target.dim\_machine\_library](#_heading=h.32hioqz) 27

[Table Name: dim\_cost\_estimation\_conversion](#_heading=h.1hmsyys) 29

[Table Name: template\_master](#_heading=h.41mghml) 30

[Table Name: target.dim\_calendar](#_heading=h.2grqrue) 31

[Table Name: dbo.template\_data](#_heading=h.vx1227) 33

[Table Name: dbo.simplifiedTimerlog](#_heading=h.3fwokq0) 33

[Table Name: dbo.uploadFilelog](#_heading=h.1v1yuxt) 35

[Table Name: dbo.simplified\_data](#_heading=h.4f1mdlm) 36

[5. System Integration and Monitoring](#_heading=h.2u6wntf) **39**

[**5.1. Error Management**](#_heading=h.19c6y18) **39**

[**5.2. Alerts and Notifications**](#_heading=h.3tbugp1) **40**

[**5.3. Version Control**](#_heading=h.28h4qwu) **40**

[**5.4. Programming Standards**](#_heading=h.nmf14n) **40**

[6. System Policies](#_heading=h.37m2jsg) **40**

[7. API Integration](#_heading=h.1mrcu09) **41**

[8. Reporting Solution (Low Level Design)](#_heading=h.2lwamvv) **76**

[9. UI Solution (Low Level Design)](#_heading=h.3l18frh) **77**

[9.1. Application Process Flow](#_heading=h.206ipza) **77**

[9.2. Page Process/Navigation Flow](#_heading=h.2zbgiuw) **77**

[10. Backend (Low Level Design) (Scripting/SQL Processing)](#_heading=h.sqyw64) **81**

[11. References](#_heading=h.1rvwp1q) **81**

[12. Appendix](#_heading=h.4bvk7pj) **82**

# Introduction

# Background / Business Problem / Business Overview

The purpose of the project is to give a business user a provision to calculate the total cost of any product from scratch. This cost is calculated by entering the relevant information about the raw materials, conversion rates and all the other costs related to the manufacturing and logistics of the product. We can also do a Should cost analysis of that product with the help of the dashboards to get a better understanding of the data.

# Purpose

This document includes all the information about the architecture of the project. It explains in detail the resources and services used, the data model, functionalities, workflow. It describes the major components that will make up the solution, the dependencies between them, and

how they will work together.

# Related Documents

| **Sr.No.** | **Document Name** | **Location / Link** |
| --- | --- | --- |
| 1. | Release notes:   * 29th Sept 2022 * 12th Oct 2022 * 4th Nov 2022 |  |
| 2. | Backend code setup |  |
| 3. | Frontend code setup |  |
| 4. | User manual |  |

# 

# Scope of Work

# Front End: Visualizations / UI / Dashboard

* + 1. **Application**

The frontend of the application is completely based on ReactJS. It has the should cost creation tool and different libraries/modules required. It also has a summary page which gives a brief understanding of the project with the help of a chart and a waterfall graph.

* + 1. **Dashboard**

The dashboards are created in Tableau. Various visualizations like Claw-back quantifiers (Category View, Vendor View), Input Factor Analysis (by SKU, Category, Vendor), Early Warning Radar, Cost Breakdown by Vendor are available.

| **Name of component** | **Short Description** | **Reference for mockup** |
| --- | --- | --- |
| Build should cost | List of all the products with the details and status whether the should cost is calculated or not. |  |
| Modify should cost | Shows all the should cost projects. Has an option to delete or modify a project. It has filters which can be used to filter the projects. |  |
| Machine library | Has the data about all the machines used for manufacturing, packaging, etc. of a material. |  |
| Template library | Should cost can also be calculated using a premade template which has all the required data. |  |
| Conversion library | Has the data about all the conversion rates, currencies, machine conversion rates, labor, logistics, etc. |  |
| Tableau: Claw-back Quantifier (Category View/Vendor View) | Should Cost Analysis by Category as well as by Vendors, showing Projected Vendor Cost, Estimated Should Cost and difference between vendor cost and should cost. |  |
| Tableau: Country Comparison | SKU wise and country of origin wise Should Cost amount based on cost category and cost element. |  |
| Tableau: Input Factor Analysis By SKU/By Category/By Vendor | Different charts and tables for Article Details, PO Cost vs Adjusted Should Cost, Estimated Should Cost, Raw Material Variation, Conversion Cost Variation and Labor Cost Variation. |  |
| Tableau: Input Cost Radar | Input Cost Radar shows Commodity Variations and Basket Trends based on Price Variation Index. |  |
| Tableau: Commodities | Commodity Trends and Trend Details shows in this. |  |

# Back End: Data Integration / Storage

The backend of the application is based on NodeJS. We have respective APIs for each functionality. These APIs are created in NodeJS and consist of SQL queries. The queries interact with the MS SQL Server database which has all the data of the application. The NodeJS APIs are deployed as functions on the Azure function app.

| **Name of component** | **Short Description** | **Reference (If applicable)** |
| --- | --- | --- |
| ETL jobs, Scripts, Stored Procedures, Tables, Views, Layers. |  |  |
|  |  |  |

# Out of Scope

* Scenario comparison – Comparing different should cost projects created for the same product and country within the application.
* Deployment pipeline restructuring.

# Dependencies/Constraints/Risk/Gaps

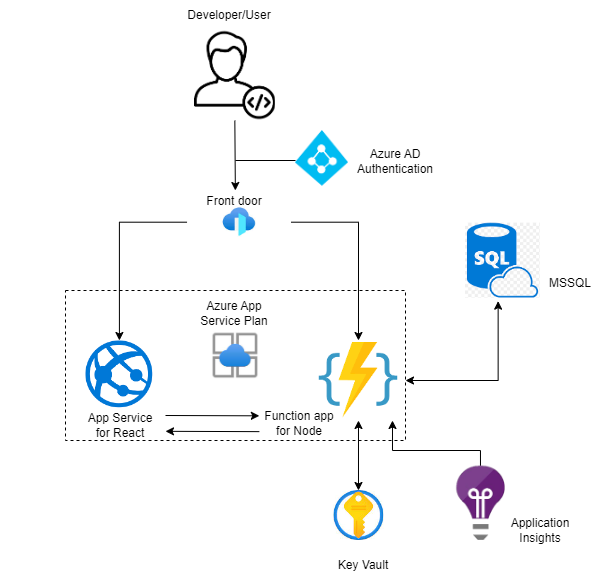
Need to restructure and clean the data model because it slows down the interaction process of the application.

# Definitions

| **Term / Abbreviation** | **Description** |
| --- | --- |
| O | Optional |
| M | Mandatory |
| NA | Not applicable |

# 

# Architecture Overview



# Understanding of Current System Architecture

The current system of the application has an end user who has access to the ICE application. Using this, he can create a should cost of any product for any country from scratch. It is a React based application which is integrated with an MS SQL database.

The backend of the architecture is NodeJS. There are APIs created for every functionality. These APIs are deployed on the Azure function app as azure functions.

One can also refer to the Dashboards for the ICE application. On the Homepage of the application, there are links provided which redirect you to the dashboards.

# Architecture

## **System Characteristics**

The description of the system should be given in terms of the architecture of the solution that is being implemented with high level data flows described to set the context of the system, i.e. to look at its external interfaces. This section should also set out to ‘characterise’ the system describing aspects of its operation that indicate if the system has:

* To operate in real-time or in bursts, for example, linked to month-end reporting
* The nature of the interface to the users of the system
* Number of concurrent users (e.g. number of maximum user logged in system at the same time.)
* To be highly resilient or fault tolerant.
* To provide security features to protect data.
* To be scalable and easily maintainable in the future.

## **Process Flow**

Timeline

Description automatically generated

## **Environment/System Configuration**

Please elaborate on machine / system configuration. These are typically ‘development’ machines.

| **Sr. No** | **IP** | **Public DNS (IPv4)** | **vCPU** | **Memory(GiB)** | **EBD Volume(GiB)** | **Node Name** |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |

Please mention IDEs to be used with exact version number.

Please mention all account / credentials information required to login. (**Please do not mention passwords**)

Mentioned details of all administrators (All applications)

| **Name of Administrator** | **User ID** | **Email ID / Phone #** | **Department.** |
| --- | --- | --- | --- |
|  |  |  |  |

## **Technology Stack**

| **Name of component** | **Version** | **Comments** |
| --- | --- | --- |
| ReactJS | v17.0.0 | It requires v16.0.0 for local development |
| NodeJS | v14.17.3 |  |
| REST API |  | HTTP requests to GET, PUT, POST and DELETE data. |
| Azure Function app |  |  |
| Azure App service |  |  |
| Azure Active Directory |  |  |
| SQL Server |  | Database of the application |
| Tableau | v2022.1.8 |  |

## **Data Model**



Need to add updated data model

## **Detailed data model**

Below is the detailed explanation of all the tables in the ICE database.

### Table Name: Dashboard\_tableau

Text

Description automatically generated with low confidence

**This table contains the tableau data of the dashboard id**: auto increment id

**display\_name**: name of tool **description**: description of tool **link**: url of the tool

**img\_url**: stores image url of the icon **status**: status indicating if active or not **created\_date**: date of creation **created\_by**: created by

**updated\_date**: date of update **updated\_by**: updated by

Table Name: target.commodities

**This table contains all the commodities and their values**

**commodity\_id**: unique id of the commodity

**date**: date of creation **value**: value of commodity **created\_by**: created by

**updated\_date**: date of updation

**updated\_by**: updated by

### Table Name: country\_mapping

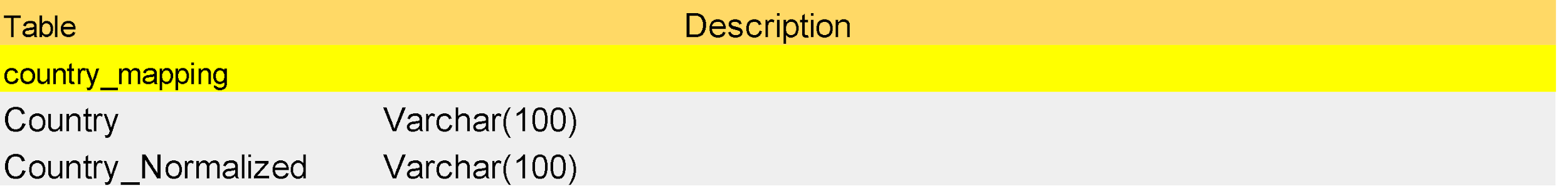


Table Name: target.dim\_tariff



**This table contains the tariff details**

**id**: auto increment id **level\_1**: industry of tariff **level\_2**: category of tariff **level\_3**: chapter of tariff **level\_4**: sub category of tariff

**level\_5**: product group of tariff

**level\_6**: product of tariff

**description**: additional description of tariff

**hts\_code**: hts code

**country**: country of origin **created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

### Table Name: dim\_project

A picture containing timeline

Description automatically generated

**This table contains the overview of the projects.**

**id**: auto increment id

**name**: name of the project **article\_id**: sku of the project **country**: country of origin **created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

**status**: contains the status of project - completed, in progress, simplified, template, simplified template

**application**: application of the project

**parent\_project\_id**: project id of parent project corresponding to id of dim\_project

### Table Name: dim\_cost\_estimation\_commodities

Text

Description automatically generated

**This table contains the cost estimations of commodities used by lookup functions**

**id**: auto increment id

**project\_id**: project id corresponding to id of dim\_project **subsection\_id**: subsection id corresponding to id of dim\_subsection **variable\_name**: name of cost estimation commodity

**value**: value of cost estimation commodity

**internal\_order**: internal order of cost estimation commodity

**created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

**section\_id**: section id corresponding id of dim\_section **commodities\_id**: unique id corresponding to id of dim\_commodities **uom**: unit of measurement

**look\_up**: look up name corresponding to index\_ticker of dim\_commodities

### Table Name: dim\_project\_cost\_details

Table

Description automatically generated

**This table contains the cost details of a project with their specification, commodities, conversion, machine, formula and tariff.**

**id**: auto increment id

**project\_id**: project id corresponding to id of dim\_project **element\_id**: element id corresponding to id of dim\_elemet **section\_id**: section id corresponding id of dim\_section **subsection\_id**: subsection id corresponding to id of dim\_subsection

**type\_id**: type id - 0: spec material, 1: commodities, 2: conversion, 3: machine, 4: formula, 5: tariff **tb\_id**: reference id for conversion\_library\_id, commodities\_id, tariff\_id, machine\_library\_id, if any **variable\_name**: name

**value**: value

**adjusted\_value**: adjusted value **uom**: unit of measurement **index\_spec**: specification **comments**: additional comments

**final\_formula**: flag to check if formula is for a final formula

**internal\_order**: internal order **created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

**is\_packaging**: flag to check if material is part of packaging

**formula**: formula used for calculation

**tb\_ref\_id**: reference id corresponding to id in cost\_estimation\_formula

**items\_names**: items used in formula

**subsection\_formula**: flag to check if formula is subsection formula

**should\_cost**: should cost of project

**linked\_sku**: linked sku of project

### Table Name: target.cost\_estimation\_summary

Text

Description automatically generated

**This table contains the cost estimation summary of the project.**

**section\_id**: section id corresponding id of dim\_section

**subsection\_id**: subsection id corresponding id of dim\_subsection

**index\_ticker**: unique index

**value**: value of cost estimation summary **fiscal\_year**: contains the fiscal year **fiscal\_month**: contains the fiscal month

**project\_id**: project id corresponding id of dim\_project

**simplified\_project\_id**: linked simplified project id

**date**: date of project creation

### Table Name: target.dim\_spec\_material

| target.dim\_spec\_material | | Contains specification data |
| --- | --- | --- |
| id | PK, int, not null |  |
| element\_id | FK, int, not null |  |
| specification | varchar(50), not null |  |
| uom | varchar(50), null |  |
| additional\_description | varchar(100), null |  |
| internal\_order | int, null |  |
| variable\_name | varchar(50), null |  |
| value | decimal(18,4), null |  |
| created\_date | date, null |  |
| created\_by | varchar(100), null |  |
| udpated\_date | date, null |  |
| udpated\_by | varchar(100), null |  |
| subsection\_id | FK, int, null |  |

**This table contains the specification data of materials of the project. id**: auto increment id

**element\_id**: element id corresponding to id of dim\_elemet

**specification**: type of specification

**uom**: unit of measurement of specification **additional\_description**: any additional description of specification **internal\_order**: internal order of specification

**variable\_name**: name of specification **value**: value of specification **created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

**subsection\_id**: subsection id corresponding id of dim\_subsection

### Table Name: target.dim\_subsection

**This table contains the subsection details of projects. id**: auto increment id

**element\_id**: element id corresponding to id of dim\_elemet **section\_id**: section id corresponding id of dim\_section **name**: name of subsection

**internal\_order**: internal order of subsection

**created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

### Table Name: target.dim\_conversion\_library

**This table contains the conversion library data.**

**id**: auto increment id

**country**: country of origin

**industry**: industry conversion belongs to **application**: application of conversion **name**: name of conversion

**process**: process of conversion **description**: description about conversion **value\_final**: final value of conversion **uom**: unit of measurement

**comments**: additional comments about conversion

**data\_source**: source of data **created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

### Table Name: target.baseline\_costs

**This table contains PO for creating simplified projects. fiscal\_year**: contains the fiscal year

**fiscal\_month**: contains the fiscal month **article**: sku of project **country\_of\_origin**: country of origin **merchant\_cat\_id**: merchant cat id **vendor\_id**: vendor id

**cost**: cost of PO

**created\_date**: date of creation

**created\_by**: created by

### 

### Table Name: target.dim\_machine\_library

Text

Description automatically generated with medium confidence

**This table contains the machine library data for projects.**

**id**: auto increment id

**name**: name of machine

**details**: additional details of machine **country**: country of origin **application**: application of machine **machine\_cost**: cost of machine **lifespan**: lifespan of machine

**throughput\_minute**: throughput per minute of machine **energy\_consumption\_kwh**: energy consumption of machine in kwh **operation\_FTEs\_per\_line**: operation FTEs per line of machine **supervisor\_FTEs\_per\_line**: supervisor FTEs per line of machine **scrap**: scrap generated by machine

**attribute**: attribute of machine

**hours\_per\_day**: utilization of machine hours per day

**days\_per\_week**: utilization of machine days per week

**created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

### Table Name: dim\_cost\_estimation\_conversion



**This table contains the conversion cost estimation data.**

**id**: auto increment id

**project\_id**: project id corresponding to id of dim\_project **section\_id**: section id corresponding id of dim\_section **subsection\_id**: subsection id corresponding subsection id **conversion\_library\_id**: corresponding conversion\_library\_id **adjusted\_value**: adjusted value of conversion **internal\_order**: value of order

**created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by **variable\_name**: name of conversion

**uom**: unit of measurement of conversion

**value**: value associated with lookup in dim\_conversion\_library table

**lookup\_name**: lookup name associated with dim\_conversion\_library table

Table Name: template\_master



**This table contains all the data of the projects displayed in the template library. id**: auto increment id

**proj\_id**: id of the project corresponding to id of dim\_project

**country**: country of the project **category\_group**: category group of the project **category**: category of the project **sub\_category**: sub category of the project **description**: description of the project

**status**: status of the project: Simplified, complex

**created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

### Table Name: target.dim\_calendar

A picture containing timeline

Description automatically generated

**This table contains the data in cost estimation summary table. id:** auto increment id

**date**: contains date in **YYYY-MM-DD** format **calendar\_year**: contains calendar year **calendar\_month**: contains month of the year (numeric) **calendar\_month\_name**: contains name of the month **fiscal\_year**: contains the fiscal year

**fiscal\_quarter**: contains the quarter number

**fiscal\_week**: contains week number

**fiscal\_month\_first\_date**: contains date of the beginning of fiscal month

**fiscal\_year\_label**: contains fiscal year label (eg: FY20)

**day\_of\_week**: contains day of the week

### Table Name: dbo.template\_data

| **Table** | | **Description** |
| --- | --- | --- |
| dbo.template\_data | | using this table for dumping raw template data from excel |
| id | PK, int, not null | auto increment id |
| data  status | varchar(max), null  varchar(50), null | contain single template row in string format  status indicate project create or not, different status are (not started, in progress, completed, failed) |
| created\_date | date, null | date creation date |
| created\_by | varchar(100), null | date created by |
| updated\_date | date, null | data updation date |
| updated\_by | varchar(100), null | data updated by |
| log | varchar(1000), null | contain error message if failed to create project |

**This table is used for dumping raw template data of complex projects from excel id**: auto increment id

**data**: contain single template row in string format

**status**: status of should cost calculation: not started, in progress, completed, failed

**created\_date**: date of creation **created\_by**: created by **updated\_date**: date of updation **updated\_by**: updated by

**log**: contains error message if should cost calculation fails

### Table Name: dbo.simplifiedTimerlog



**This table is used for tracking the should cost calculation of simplified projects. id**: auto increment id

**project\_id**: id of the project corresponding to id of dim\_project

**should\_cost**: value of should cost

**error**: records error if calculation fails

**executionTime**: time taken for should cost calculation

**createdAt**: time of data entry in this table

### Table Name: dbo.uploadFilelog

**This table is used for tracking simplified projects upload in db.**



**id**: auto increment id

**userid**: track user id that uploaded the data

**request**: request payload sent to backend

**error**: records error if calculation fails

**uploadedDataLength**: number of rows in the uploaded file **rowsAffected**: number of rows added to db (dbo.simplified\_data table) **ignoredDuplicateRows**: number of duplicate entries ignored **executionTime**: time taken data upload in milliseconds

**createdAt**: time of data entry in this table

### Table Name: dbo.simplified\_data

**This table is used for dumping raw simplified data from excel. id**: auto increment id

**serial\_no**: sku of the project

**vendor**: vendor id

**item\_name**: name of item in project **industry**: industry of the project **category**: category project belongs to

**description**: description of the project

**origin**: country of origin

**name**: name of cost details ex: labor cost, profit, etc.

**value**: cost corresponding to name

**status**: maintains status of should cost calculation: not started, completed, failed, in progress

**created\_date**: date of creation

**created\_by**: created by

**updated\_date**: date of updation

**updated\_by**: updated by

Table Name: target.dim\_commodities

**This table contains details of the commodities library.**

**id**: auto increment id

**level\_1**: level 1

**level\_2**: level 2

**level\_3**: level 3

**description**: description of commodity

**index\_tracker**: unique index corresponding to index\_ticker of dim\_commodities

**units**: unit of measurement

**level\_4**: level 4 **data\_source**: source of data

**created\_date**: date of creation **created\_by**: created by

# System Integration and Monitoring

## **Error Management**

Errors occurring on the application are logged into a log file at the backend. Below attached is a JSON format in which the errors get logged.

**{**

**"level": "error",**

**"message": {**

**"controller": "getProjectSummaryTable>TableCostSummary.getPackagingSummary",**

**"error": {}**

**},**

**"timestamp": "2022-11-10 12:44:11"**

**}**

* + 1. **Deployment error trackbacking**

**Method 1:** Deployment of the frontend and backend code can sometimes be unsuccessful. To trace back this, you can view the zip folder created as an artifact. If this zip folder doesn’t have all the data you have uploaded through the git branch then you will need to check the code if there are any commits done and rerun the build pipeline.

**Method 2:** All the APIs for the application must be passed through Redux in order to work on the client-side application.

**Method 3:** If there are any changes in the azure functions or if any new azure function is added then it should be reflected in the Azure function app on the client-side Azure portal. If this doesn’t follow, then it means there are issues with the Frontend or Backend code.

## **Alerts and Notifications**

There are alerts on the ICE application which gets displayed on the top right corner of the screen whenever any API has an error related to the database or in the backend code.

| **Condition** | **Severity** | **Alert Type (Email / Text)** | **Alert to with Description** |
| --- | --- | --- | --- |
| Error in DB | Medium | Toast on the application | Errors in the backend code related to the database |

## **Version Control**

**Azure DevOps:**

1. Repositories:
   1. Frontend - [ICE - Frontend Repo](https://dev.azure.com/gto-prime/GTO-ICE/_git/gto-ice-frontend?path=/&version=GBdevelopment)
   2. Backend - [ICE - Backend Repo](https://dev.azure.com/gto-prime/GTO-ICE/_git/gto-ice-backend?path=/&version=GBmain)
2. Boards:

<https://dev.azure.com/gto-prime/GTO-ICE/_boards>

# System Policies

* 1. **Security / Governance**

**Authentication:** Once the user is logged in, the server generates a JWT token, each subsequent request will include the JWT, allowing the user to access routes, services, and resources that are permitted with that token.

Proper token invalidates mechanism to mark them expire as and when required.

# API Integration

# This document describes all the APIs that are involved in the Application. The payloads, responses, results of the APIs are mentioned in the document.

1. **Conversion API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/con versions | Post | req.body | {"country":"Uk","industry ":"Consumables","applic ation":"Chair","name":"s ad","index":"d","descript ion":"d","value\_final":2," uom":"d","comments":"d "} | "status": true, "message": "Data created successfully", "data": { "rowsAffected": [ 1 ], "row": { "country": "Uk", "industry": "Consumables", "application": "Chair", "name": "sad", "index": "d", "description": "d", "value\_final": 2, "uom": "d", "comments": "d", "created\_date": "2021-12-17T00:00:00.000Z", "created\_by": "admin", "updated\_date": "2021-12-17T00:00:00.000Z", "updated\_by": "admin", "id": 1269 } } | **A**PI for creating conversion |
| /api/v1/get Conversions | Request  Type:Post | req.body | { page : 1, country : "Asia" , industry :"Electronics", application : "PCBA Component", text :"Asia" } | "status": true, "message": "Data retrieved successfully", data":{ "records": [ {"id": 281,"country": "Asia", "industry": "Electronics", "application": "PCBA Component","name": "WeEn, Part #: ACT108W-800E", "process": "ACT108W-800E", "description": "TRIAC, 800V 0.8A SOT-223", "value\_final": 0.07, "uom": "$/Each", "comments": null, "data\_source": "Internal", "created\_date": null, "updated\_date": null }, ], "pagination": { "total\_records": 234, "page\_size": 10, "page\_no": 1 } } | API for getting conversion |
| /api/v1/conversions  /:conversionId | Request Type: GET | Path parameters | NA | "status": true, "message": "Data retrieved successfully", "data": [{ "id": 356, "country": "Asia", "industry": "Electronics", "application": "PCBA Component", "name": "Vishay, Part #:VDRS14T275TyE", "process": "ass", "description": "Jumper,15mm  length", "value\_final": 0.03, "uom": "$/Each", "comments": "sa", "data\_source": "Internal", "created\_date": “2021-10-30”, "created\_by": admin, "updated\_date": "2021-10-30", "updated\_by": "admin" } ] | API for getting specific conversion |
| /api/v1/conversion/countries | Request Type : GET | NA | NA | "status": true, "message": "Data retrieved successfully", "data": [ { "country": "AFGHANISTAN" },{ "country": "NIUE" }, { "country": "NORFOLK ISLAND" }, { "country": "NORWAY" }, { "country": "OMAN" }, { "country": "PERU" }, { "country": "PHILIPPINES" }, { "country": "PITCAIRN ISLANDS" }, { "country": "POLAND" }, { "country": "PORTUGAL" }, { "country": "QATAR" }, { "country": "SAN MARINO" }, { "country": "SAO TOME AND PRINCIPE" }, { "country": "SAUDI ARABIA" }, { "country": "SENEGAL" }, { "country": "SERBIA" }, { "country": "SEYCHELLES" },] | API for getting countries |
| /api/v1/conversion/ind ustries/:country | Request Type : GET | NA | NA | "status": true, "message": "Data retrieved successfully", "data": [ { "industry": "Consumables" }, { "industry": "Manufacturing" }, { "industry": "s" } ] | API for getting industries |
| /api/v1/conversion/applications/:country/:industry | Request Type : GET | NA | NA | "status": true, "message": "Data retrieved successfully", "data": [ { "application": "PCBA Component" } ] | API for getting application |
| /api/v1/conversions/:conversionId | Request Type : Post | req.body | {country:"Uk" , industry:"Consumabl es",application:"Ch air",name:"sad",ind ex:"d",description: "d",value\_fnal:2, uom:"d",comments:"d "} | "status": true, "message": "Data updated successfully for record id = 1", "data": { "id": 1, "country": "Uk", "industry": "Consumables", "application": "Chair", "name": "sad", "index": "d", "description": "d", "value\_final": 2, "uom": "d", "comments": "d" } | API for updating conversion |
| /api/v1/conversion/delete | Request Type : Post | req.body | {"conversionId":356} | { "status": true, "message": "Data deleted successfully" } | This API is used when a user deletes conversion. |

1. **Formula item**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/finalFormula | Request Type : Post | req.body | {"project\_id":122 ,"element\_id":154 ,"section\_id":1," element\_name":"New"} | { "status": true, "message": "Data created successfully" } | API for creating final formula |
| /api/v1/formulaItem | Request  Type : Post | req.body | {"project\_id":1347,"typ  e":"other\_cost"} | "status": true, "message":  "Data retrieved successfully",  "data": { "records": [ { "id": 46,  "project\_id": 16, "section\_id":  2, "subsection\_id": 21,  "machine\_library\_id": 67,  "Yogurt production  (Combined)", "variable\_name":  "e11\_Machine\_lifespan",  "adjusted\_value": 20,  "internal\_order": 1, "uom":  "Yrs", "value": 20 }],  "totalCost": 662.15 } | API for getting  formula items |

1. **Element API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/elem  ent | Request Type : Post | req.body | { "project\_id":2,  "name":"te23st",  "internal\_order":1,  "packaging\_flag":0} | "status": true, "message":  "Data created successfully",  "result": { "recordset": {  "project\_id": 2, "name":  "te23st", "packaging\_flag":  0 } } | API for  creating  element |
| /api/v1/updat  eElement | Request  Type : Post | req.body | {"element\_id":10,  "project\_id":31,  "name":"Test12345",  "packaging\_flag":1} | "status": true, "message":  "Data updated successfully", "result": {"recordset":  { "element\_id":10,  "project\_id": 31,  "name": "Test12345",  "packaging\_flag": 1 } } | API for  updating  element |
| /api/v1/getEl  ement | Request  Type : Post | query  params | {"project\_id":1} | "status": true, "message":  "Data retrieved  successfully", "data": [ {  "id": 331, "project\_id": 361,  "name": "Wrap\_2",  "internal\_order": 12,  "packaging\_flag": false,  "machine\_library\_id": null,  "machine": [] }, { "id": 329,  "project\_id": 361, "name":  "Wrap\_1", "internal\_order":  12, "packaging\_flag": true,  "machine\_library\_id": null,  "machine": [] } ] | API for getting  elements |

1. **Landing Page API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/landing/ah3 | Request  Type : GET | NA | NA | "status": true,  "message": "Data retrieved  successfully",  "data": [ { "ah3": "HOUSEHOLD  GOODS" } ] | API for  getting ah3 |
| /api/v1/landing/ah5 | Request  Type : Post | NA | NA | "status": true, "message": "Data retrieved  successfully", "data": [ { "subcategory\_ah5":  "BATH TISSUE" }, { "subcategory\_ah5": "CUPS"  }, { "subcategory\_ah5": "CUTLERY" }, {"subcategory\_ah5": "DESIGNER TABLEWARE" },  { "subcategory\_ah5": "FACIAL TISSUE" }, {  "subcategory\_ah5": "FLUSHABLE WIPES" }, {  "subcategory\_ah5": "FOIL" }, {  "subcategory\_ah5": "MISCELLANEOUS" }, {  "subcategory\_ah5": "NAPKINS" }, {  "subcategory\_ah5": "PAPER TOWELS" }, {  "subcategory\_ah5": "PLATES/BOWLS" }, {  "subcategory\_ah5": "TRASH BAGS" }, {"subcategory\_ah5": "WRAPS" } ] | API for  getting ah5 |

1. **Build Should Cost Screen -API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/landi  ng/rows | Request  Type : GET | query params | page,group,  category,sub  category,text | "status": true, "message": "Data  retrieved successfully", "data": {  "records": [ { "id": 377631,  "description": "BJ 45 GAL OUTDOOR  TRASH BAGS 100CT", "color": "TBD",  "material": "Plastic", "ah3":  "HOUSEHOLD GOODS",  "category\_ah4": "TRASH",  "subcategory\_ah5": "TRASH BAGS" }],  "pagination": { "total\_records": 41,  "page\_size": 10, "page\_no": 1 } } | API for  getting  link sku  data |
| /api/v1/build  ShouldCost/r  ows | Request  Type : GET | query params | page,group,  category,sub  category,text | "status": true, "message": "Data  retrieved successfully", "data": {  "records": [ { "id": 707383,  "description": "BERKLEY JENSEN  12/160SHEET WITH SAS TOWE",  "color": "White", "material": "Paper",  "ah3": "HOUSEHOLD GOODS",  "category\_ah4": "PAPER",  "subcategory\_ah5": "PAPER TOWELS"  } ], "pagination": { "total\_records": 6,  "page\_size": 10, "page\_no": 1 } } | API for  getting  build  should  cost data |

1. **Template Screen- API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/  Existing  Template/rows | Request  Type : GET | query  params | page,sku,group  ,category,subc  ategory,country  ,text | "status": true, "message": "Data retrieved  successfully", "data": { "records": [ { "id":  242990, "country": "EQUATORIAL GUINEA",  "status": "completed", "project\_id": 1164,  "description": "EVERYDAY NAPKINS 660CT",  "color": "TBD", "material": "Paper", "ah3":  "HOUSEHOLD GOODS", "category\_ah4":  "PAPER", "subcategory\_ah5": "NAPKINS",  "should\_cost": "655.85" }], "pagination": {  "total\_records": 6, "page\_size": 10, "page\_no": 1  } } | API for  getting  template  data |

1. **Modify should cost -API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/modifyProject/row  s | Request  Type : Post | req.body | {"page":1,  "text":"test",  "group":"group",  "category":"FOOD  STORAGE/FOIL/WRAPS",  "subcategory":"subcategory",  "country":"Asia",  "status":"" | "status": true, "message": "Data  retrieved successfully", "data": {  "records": [ { "id": 262301,  "project\_id": 3248,  "country":"LITHUANIA",  "status":"Simplified", "should\_cost":1.6579, "description":"262301\_project", "color": "SilverMetallic", "material": "Aluminum",  "ah3": "HOUSEHOLD GOODS",  "category\_ah4": "FOOD  STORAGE/FOIL/WRAPS",  "subcategory\_ah5": "FOIL",  "linked\_sku": 3246,  "dataBySection": [ {  "section\_name": "Raw Material +  Scrap", "value": 0.9748 }, {  "section\_name": "Overhead &  Margin", "value": 0.0166 }, {  "section\_name": "Conversion",  "value": 0.6499 }, {  "section\_name": "Duty / Tariff",  "value": 0.0166 } ] } ], "pagination":  { "total\_records": 2, "page\_size":  10, "page\_no": 1 } } | API for  getting  modify  project data |

1. **Commodities - API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/commodity/level1 | Request  Type : GET | NA | NA | "status": true, "message": "Data retrieved  successfully", "data": [ { "level\_1": "Chemicals" },  { "level\_1": "Economics" }, { "level\_1": "Energy &  Transport" }, { "level\_1": "Ferrous" }, { "level\_1":  "Metals & Ores" }, { "level\_1": "NonFerrous" }, {  "level\_1": "Other Materials" }, { "level\_1":  "Plastic & Textiles" }, { "level\_1": "Plastics" }, {  "level\_1": "Pulp, Paper & Wood" } ] | API for  getting  commod  ity  lookup  level1  data |
| /api/v1/commodity/level2 | Request  Type : GET | NA | NA | "status": true, "message": "Data retrieved successfully", "data": [ { "level\_2": "Aluminum" },  { "level\_2": "Art paper" }, { "level\_2": "Benzene"  },, { "level\_2":  "Kraftliner/kraft bags" },  {  "level\_2": "Label" },  { "level\_2": "Label paper" }, {  "level\_2": "Labor Rate" }, { "level\_2": "Lead" }, {  "level\_2": "Logs" }, { "level\_2": "Low density  polyethylene (LDPE)" }, { "level\_2": "Lumber" }, {  "level\_2": "Medium density fibreboard (MDF)" },  { "level\_2": "News wastepaper" }, { "level\_2":  "Newsprint" }, { "level\_2": "Non woven fabric" },  { "level\_2": "Oriented strand board (OSB)" }, {  "level\_2": "Packaging Index" }, { "level\_2":  "Packaging paper" }, { "level\_2": "Paint" }, {  "level\_2": "Paper" }, { "level\_2": "Paper and  board" }, { "level\_2": "Paper and board  corrugated material" }, { "level\_2": "Paper boxes  and containers" }, { "level\_2": "Paper  cartons/boxes" }, { "level\_2": "Paper household /  sanitary" }, { "level\_2": "Paper maplitho" }, {  "level\_2": "Paper packaging" }, { "level\_2": "Paper uncoated woodfree" }, { "level\_2": "Paperboard" }, { "level\_2": "PET" }, { "level\_2": "Plywood" }, { "level\_2": "Polyethylene (PE)" }, { "level\_2": "Polyethylene Terephthalate (PET)" }, { "level\_2": "Polypropylene" }, { "level\_2": "Polypropylene (PP)" }, { "level\_2": "Polystyrene" }, { "level\_2": "Polystyrene (PS)" }, { "level\_2": "Polyvinyl Chloride (PVC)" }, { "level\_2": "Propylene" }, { "level\_2": "Publication paper" }, { "level\_2": "Pulp" }, { "level\_2": "Pulp and paper products" }, { "level\_2": "Pulp; paper and converted paper products" }, { "level\_2": "PVC" }, { "level\_2": "Recovered (waste and scrap) paper" }, { "level\_2": "Recovered (waste and scrap) paper or paperboard" }, { "level\_2": "Recovered corrugated cartons; boxes; or sheets" }, { "level\_2": "Recycled paper" }, { "level\_2": "Rubber" }, { "level\_2": "Sawlogs" }, { "level\_2": "Sawnwood" }, { "level\_2": "Testliner" }, { "level\_2": "Thermal paper" }, { "level\_2": "Tissue paper" }, { "level\_2": "Titanium Oxide" }, { "level\_2": "Venered panels" }, { "level\_2": "Waste paper" }, { "level\_2": "Wastepaper" }, { "level\_2": "Woodchips and particles" }, { "level\_2": "Wrapping paper packaging" }, { "level\_2": "Zinc" } ] |  |
| /api/v1/commodity/level3 | Request  Type : GET | NA | NA | "status": true, "message": "Data retrieved  successfully", "data": [ { "level\_3": "Aluminum" },  { "level\_3": "Zinc" } ] | API for  getting  commod  ity  lookup  level3  data |
| /api/v1/commodity/count  ry | Request  Type : GET | NA | NA | "status": true, "message": "Data retrieved  successfully", "data": [ { "level\_4": "Asia" }, {  "level\_4": "Australia" }, ] | API for  getting  Commodity  lookup  countries  data |
| /api/v1/com  modities | Request  Type : GET | NA | level1,level2,level3,  country,  text,page | "status": true, "message": "Data retrieved  successfully", "data": { "recordset": [ { "id": 444,  "level\_1": "Plastic & Textiles", "level\_2":"Polyethylene (PE)", "level\_3": "Polyethylene  (PE)", "description": "HDPE", "series\_specific":  "HDPE", "index\_ticker": "SB13", "units":  "USD/metric tonne", "level\_4": "China",  "data\_source": "Mintec", "updated\_date":  "11/11/2021", "value": 0 }], "pagination": {  "total\_records": 382, "page\_size": 10,  "page\_no": 1 } } | API for  getting  commod  ities  lookup  data |
| /api/v1/country | Request  Type : GET | NA | NA | "status": true,  "message":  "Data retrieved successfully",  "data": [  { "name":"AFGHANISTAN" }, { "name": "ALBANIA" },  {"name": "ALGERIA" }, { "name": "ANDORRA" }, {  "name": "ANGOLA" }, { "name": "ANGUILLA" }, {  "name": "ANTIGUA AND BARBUDA" }, { "name":  "ARGENTINA" }, { "name": "ARMENIA" }, {  "name": "ARUBA" }, { "name": "ASIA" }, {  "name": "AUSTRALIA" }, { "name": "AUSTRIA" }, {  "name": "AZERBAIJAN" }, { "name": "BAHAMAS"  }, { "name": "BAHRAIN" }, { "name":  "BANGLADESH" }, { "name": "BARBADOS" }, {  "name": "BELARUS" }, { "name": "BELGIUM" }, {  "name": "BELIZE" }, { "name": "BENIN" }, {  "name": "BERMUDA" }, { "name": "BHUTAN" }, {  "name": "BOLIVIA" }, { "name": "BOSNIA AND  HERZEGOVINA" }, { "name": "BOTSWANA" }, {  "name": "BRAZIL" }, { "name": "BRITISH INDIAN  OCEAN TERRITORIES" }, { "name": "BRITISH  VIRGIN ISLANDS" }] | API for  getting  country  data |

1. **Machine API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| api/v1/mach  ines/create | Request  Type : Post | req.body | {"name":  "test","details":  "details","country":  "asia","application":  "application","machine\_cost": 100,"lifespan":  2,"throughput\_minute":10,"energy\_consumption\_Kwh":  10,"operator\_FTEs\_per\_line":10,"supervisor\_FTE\_per\_line":10,"scrap":10,"attribute":"test","hours\_per\_day":10,"days\_per\_week":10} | {"status":true,"message":"Data created  successfully","data":{"rowsAffected":[1],"  row":{"name":"test123","details":"test","country":"ALBANIA","application":"dsdf3lk","machine\_cost":3424,"lifespan":32,"throughput\_minute":23432,"energy\_consu  mption\_Kwh":124,"operator\_FTEs\_per\_line":4,"supervisor\_FTE\_per\_line":4,"scrap":0,"attribute":"23","hours\_per\_day":15,"days\_per\_week":5,"created\_date":"2022-0301","created\_by":"admin","updated\_date":"2022-03-01","updated\_by":"admin","id":77}}} | API for  creating  machine |
| /api/v1/getM  achines | Request  Type :  Post | query  param | page, country,  application, text | "status": true, "message": "Data  retrieved successfully", "data": {  "records": [ { "id": 131, "name": "TADBASED ROLLING MACHINE (USA)","details": "Process used in paper","country": "USA", "application": "Paper Towel", "machine\_cost": 800000,  "lifespan": 10, "throughput\_minute":600, "energy\_consumption\_Kwh": 150,  "operator\_FTEs\_per\_line": 3,  "supervisor\_FTE\_per\_line": 0, "scrap": 1,  "attribute": "0", "hours\_per\_day": 24,  "days\_per\_week": 6 },], "pagination": {  "total\_records": 131, "page\_size": 10,  "page\_no": 1 } } | API for  getting  machine data |
| /api/v1/machines/:machin  eId | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ { "id":  11, "name": "Plastic Injection molding","details": "Process used in household products - cups, lids etc.", "country":"USA", "application": " 240CT/18OZ  PLASTIC RED CUP","machine\_cost":  800000, "lifespan": 10,  "throughput\_minute": 300,  "energy\_consumption\_Kwh": 150,  "operator\_FTEs\_per\_line": 3,  "supervisor\_FTE\_per\_line": 0.33, "scrap":  0.01, "attribute": null, "hours\_per\_day":  24, "days\_per\_week": 6, "created\_date":  "2021-11-01T00:00:00.000Z",  "created\_by": "manual",  "updated\_date": admin, "updated\_by":  admin } ] | API for  getting  specific  machine |
| /api/v1/machine/countries | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "country": "AFGHANISTAN" }, { "country":  "ALBANIA" }, { "country": "ALGERIA" }, {  "country": "ANDORRA" }, { "country":  "ANGOLA" }, { "country": "ANGUILLA" }, {  "country": "ANTIGUA AND BARBUDA" }, {  "country": "ARGENTINA" }, { "country":  "ARMENIA" }, { "country": "ARUBA" }, {  "country": "AUSTRALIA" },, { "country":  "BAHRAIN" }, { "country": "BULGARIA" },  { "country": "BURKINA FASO" }, {  "country": "BURMA" }] | API for  getting  Machine  countries |
| /api/v1/machine/applicatio  ns/:country | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "application": "Drinks" }, { "application":  "dsdf3lk" }, { "application": "Paper towel" } ] | API for  getting  Machine  application |
| /api/v1/mach  ine/delete | Request  Type :  POST | req.body | {"machineId":9596} | {"status":true,  "message":  "Data deleted successfully"} | API for  deleting  Specific  machine |

1. **Project Details API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/project | Request Type : Post | Req.body | {"name":"test@1$1  2",  "article\_id":1,  "country":"test"} | "status": true, "message": "Data created successfully", "data": { "recordset": {  "project\_id": 3229,  "name": "1000CT",  "article\_id": 331248,  "country": "INDIA",  "status": "In Progress", "industry": null,  "application": null, "created\_by":  "admin", "updated\_by": "admin",  "conv1": { "recordset": {  "data\_inserted\_id": 496, "project\_id":3229, "section\_id": 4, "subsection\_id":null, "conversion\_library\_id": 1,  "adjusted\_value": 0, "created\_by":"admin", "updated\_by": "admin",  "variable\_name": "margin",  "lookup\_name": "Margin % COGS",  "uom": "%", "value": 0 } }, "conv2": {  "recordset":  { "data\_inserted\_id": 497,  "project\_id": 3229,  "section\_id": 4,  "subsection\_id": null,  "conversion\_library\_id": 1,  "adjusted\_value": 0, "created\_by":"admin", "updated\_by": "admin",  "variable\_name":"corp\_overhead",  "lookup\_name": "C. OH % COGS", "uom":"%", "value": 0 } }, "conv3": { "recordset":  { "data\_inserted\_id": 498, "project\_id":  3229, "section\_id": 4, "subsection\_id":  null, "conversion\_library\_id": 1,  "adjusted\_value": 0, "created\_by":  "admin", "updated\_by": "admin",  "variable\_name": "plant\_overhead",  "lookup\_name": "2", "uom": "%", "value":  0 } }, "form1": { "recordset": {  "data\_inserted\_id": 2813, "project\_id":  3229, "section\_id": 4, "subsection\_id":  null, "element\_id": null, "formula\_name": "overhead\_&\_margin\_cost", "formula": "\"[\\\"total\_cost\\\",\\\"\*\\\",\\\"(\\\",\ \\"corp\_overhead\\\",\\\"+\\\",\\\"plant \_overhead\\\",\\\"+\\\",\\\"margin\\\",\\ \")\\\"]\"", "value": 0, "finalFormula": 1, "subsection\_formula": null, "item": "total\_cost,corp\_overhead,plant\_overhea d,margin", "created\_by": "admin", "updated\_by": "admin" } }, "tariff1": { "recordset": { "data\_inserted\_id": 198, "project\_id": 3229, "section\_id": 6, | for  creating  Project |
| /api/v1/proje  ctUpdate | Request  Type : Post | req.body | {"proj\_id":1326,  "name":"test38458",  "article\_id":707383,  "country":"VIETNAM"} | "status": true, "message": "Data updated  successfully", "data": { "recordset": {  "proj\_id": 1326,  "name": "test38458",  "article\_id": 707383,  "country":"VIETNAM",  "status": "In Progress",  "industry": null,  "application": null,  "created\_by": "admin", "updated\_by":"admin" } } | for  updating  project  details |
| /api/v1/project/updateSta  tus | Request  Type : Post | req.body | {"project\_id":16} | "status": true, "message": "Data updated  successfully" | for  updating  status |
| /api/v1/proje  ctData | Request  Type : Post | req.body | {"type":"1","project  \_id":3123,"section  \_id":1,"variable\_na  me":"test","value":  1,"commodities\_id  ":439,"uom":"USD/  metric  tonne","subsection  \_id":6157},{"type":  0,"element\_id":1,"  specification":"test  12345610","uom":  "kg","additional\_d  escription":1,"varia  ble\_name":1,"valu  e":1},{"type":"2","el  ement\_id":972,"pr  oject\_id":860,"sect  ion\_id":1,"conversi  on\_library\_id":369,  "adjusted\_value":1  3.5,"variable\_nam  e":"Direct  Labor","uom":"EU  R/hr","subsection\_  id":3802},{"type":4,  "element\_id":1133,  "project\_id":1224,"  section\_id":1,"form  ula\_name":"111sa","formula":"[\"C19\  "]","value":1,"final\_  formula":1,"item":"  C19"},"type":"1","p  roject\_id":1336,"se  ction\_id":1,"variabl  e\_name":"s8dk","v  alue":1,"commoditi  es\_id":442,"uom":"  USD/metric  tonne","subsection  \_id":4811,"lookup\_  name":"abc123"},{  "type":"5","project  \_id":1336,"section  \_id":6,"tariff\_id":3,"  value":1,"adjusted  \_value":1,"variable  \_name":"3a","uom  ":"%","lookup\_nam  e":"abc123"} | "status": true, "message": "Data created  successfully", "data": { "recordset": {  "data\_inserted\_id": 897, "project\_id": 15,  "section\_id": 1, "subsection\_id": 230,"element\_id": 79, "formula\_name":"test9", "formula":"\"[\\\"Paper\_Machine\_lifespan\\\",\\\"+\\\",\\\"Paper\_Machine\_machine\_cost\\\  "]\"", "value": 7734508, "finalFormula":null, "subsection\_formula": null, "item":"Paper\_Machine\_lifespan,Paper\_Machine\_machine\_cost", "created\_by": "admin",  "updated\_by": "admin" } } | API for  creating  project  data such  as  specificati  on,  commodit  ies,  conversion,  machine ,  formula  and tariff. |
| /api/v1/updateProjectData | Request  Type : Post | req.body | {"spec\_id":553,"ty  pe":0,"project\_id":  728,"element\_id":  796,"specification"  :"length","uom":"k  g","additional\_des  cription":"test!","va  riable\_name":"SB  11",value":9},  {"type":1,"estimati  on\_commodities\_i  d":  633,"project\_id":8  13,"subsection\_id"  :3601,"variable\_na  me":"com","value":  121,"internal\_orde  r":1,"section\_id":1,  "commodities\_id":  442,"uom":"m"}  { "type":2,  "estimation\_conve  rsion\_id":282,"proj  ect\_id":846,  "section\_id":1,  "subsection\_id":37  20,  "conversion\_librar  y\_id":366,  "adjusted\_value":1  0,  "variable\_name":"  conv1",  "uom":"kg"}, {"type":4,  "estimation\_formul  a\_id":1111,  "project\_id":853,  "element\_id": 964,  "section\_id":1,  "value":10,  "formula":"\"[\\\"sp  ec\\\",\\\"+\\\",\\\"10  0\\\"]\"",  "formula\_name":"f  orm",  "subsection\_id":37  57}, { "type":3,  "estimation\_machi  ne\_id":5921,  "project\_id":855,  "section\_id":2,  "subsection\_id":37  58,  "machine\_library\_i  d":131,  "machine\_library\_f  ield":"TAD BASED  ROLLING  MACHINE (USA)",  "variable\_name":"  e1\_Machine\_lifes  pan",  "adjusted\_value":3  0, "value":10  },{ "type":5,  "est\_tariff\_id":130,  "project\_id":855,  "section\_id":6,  "variable\_name":"t  ar1", "tariff\_id":7,  "uom":"%",  "value":1,  "adjusted\_value":1  00} | "status": true, "message": "Data updated  successfully" | API for  updating  project  data such  as  specificati  on,  commodit  ies,  conversio  n ,  machine ,  formula  and tariff. |
| api/v1/finalFor  mula | Request  Type : Post | req.body | {"project\_id":122,"  element\_id":154,"s  ection\_id":1,"elem  ent\_name":"New"} | "status": true, "message": "Data created  successfully" | API for  creating  final  formula |
| /api/v1/proje  ctDataDelete | Request  Type : Post | req.body | {type":5,  "id":100},{type":4,  "id":100}{type":3,  "id":100}{type":2,  "id":100}{type":1,  "id":100}{type":0,  "id":100} | "status": true, "message": "Data deleted  successfully", "data": { "recordset": 0 } | API for  deleting  any  specific  data (  specificati  on,  commodit  ies,  conversio  n ,  machine ,  formula  and tariff.) |
| /api/v1/sectionDetails/:proj  ectId/:elementId/:sectionId | Request  Type : GET | params | EG:  /api/v1/sectionDet  ails3123/1766/2 | "status": true, "message": "Data retrieved  successfully", "data": { "subsection": [ {  "id": 529, "element\_id": 167,  "section\_id": 1, "name": "rm1",  "internal\_order": 1, "commodities": [],  "conversion": [ { "id": 412, "project\_id":  133, "section\_id": 1, "subsection\_id":  529, "conversion\_library\_id": 282,  "value": 15.9, "adjusted\_value": 15.9,  "internal\_order": 1, "variable\_name":  "Injection Molding", "lookup\_name":  "Press Rates (3000 lb)", "uom": "$/hr" } ],  "machine": [], "formula": [ { "id": 825,  "project\_id": 133, "section\_id": 1,  "subsection\_id": 529, "formula\_name":  "rm1\_formula", "formula":  "\"[\\\"test\_Machine\_lifespan\\\"]\"",  "value": 20, "subsection\_final\_formula":  null, "subsection\_formula": 1,  "items\_names": "test\_Machine\_lifespan",  "internal\_order": 1, "uom": "%" } ],  "tariff": [] } ], "subsectionFinalFormula": {  "id": 826, "project\_id": 133, "section\_id":  1, "subsection\_id": null, "formula\_name":  "test\_raw\_material+scrap\_formula",  "formula": "'\"[\\\"rm1\_formula\\\"]\"'",  "value": 20, "subsection\_final\_formula":  1, "subsection\_formula": null,  "items\_names": "rm1\_formula",  "internal\_order": 1, "uom": "%" } } | API for  getting  particular  section  data |
| /api/v1/updateSubSection  Details | Request  Type : Post | req.body | {name:test,section  Id:2,subsectionId:  6154,elementId:17  66} | {"status":true,"message":"Data updated successfully","data":  {"recordset":{"id":12  65,"element\_id":392,  "section\_id":1,  "name":"fds",  "internal\_order":1,  "updated\_date":"2022-3-1",  "updated\_by":"admin"}}} | API for  updating  subsection |
| /api/v1/subs  ection | Request  Type : Post | req.body | {"project\_id":3114,  "element\_id":1750  ,"section\_id":2,"na  me":"23\_Machine"  ,"type":1,"machine  \_id":622} | "status": true, "message": "Data created successfully",  "data": { "recordset": {  "subsection\_id": 260, "element\_id": 4,  "section\_id": 2, "name":  "new123\_Machine" } } | API for  creating  subsection |
| /api/v1/proj  ect/unlink/:p  rojectId | Request  Type : GET | NA | NA | "status": true, "message":  "Data deleted successfully" | API for  unlinking  the  project |

1. **Tariff API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/tariff/level1 | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "level\_1": "Wood & Pulp Products" }, {  "level\_1": "Computer & Electronics" }] | API for getting  tariff lookup  level1 data |
| /api/v1/tariff/level2 | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "level\_2": "Iron and steel products" }, {  "level\_2": "Lead and lead products" }, {  "level\_2": "Wood and misc products"  }, { "level\_2": "Footware" }] | API for getting  tariff lookup  level2 data |
| /api/v1/tariff/level3 | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "level\_3": "Chapter 29 - Organic  chemicals" }, { "level\_3": "Chapter 02 -  Meat and edible meat offal" } ] | API for getting  tariff lookup  level3 data |
| /api/v1/tariff/level4 | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "level\_4": "4007 - Vulcanised rubber thread and cord" },  { "level\_4": "4704 -Chemical wood pulp,sulphite, other  than dissolving grades" }, { "level\_4":"8540 - Thermionic, cold cathode orphoto-cathode valves and tubes (e.g.vacuum, vapour, gas filled valves and  tubes, mercury arc rectifying valvesand tubes, cathode-ray and televisioncamera tubes)" }] | API for getting  tariff lookup  level4 data |
| /api/v1/tariff/level5 | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "level\_5": "500710 - Silk; woven  fabrics of noil silk" }, { "level\_5":  "120799 - Oil seeds and oleaginous  fruits; n.e.c. in heading no. 1207,  whether or not broken" }, { "level\_5":  "220870 - Liqueurs and cordials" }] | API for getting  tariff lookup  level5 data |
| /api/v1/tariff/level6 | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "level\_6": "61019010 - Mens or boysovercoats, carcoats, etc., of tex mats(other than wool, cotton or mmf), cont70% or more wt of silk, knitted orcrocheted" }, { "level\_6": "85045080 -Other inductors, nesoi" }, { "level\_6":  "91144080 - Plates and bridges for clocks, nesi" }] | API for getting  tariff lookup  level6 data |
| /api/v1/tariff/country | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "country": "Indonesia" }, { "country":  "Canada" }, { "country": "China" } ] | API for getting  tariff lookup  country data |
| /api/v1/tariff/hts | Request  Type : GET | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ {  "hts\_code": "7403130000" }, {  "hts\_code": "9506594040" } ] | API for getting  tariff lookup  hts data |
| /api/v1/tariff | Request  Type : GET | query  params | level\_1:  test,level\_2:  Mechanical  And machinery,leve  l\_3: Chapter75 - Nickel and  articles thereof  , level\_4: 3920  - Plastics;  plates, sheets,  film, foil and  strip (not  self-adhesive);  non-cellular  and not  reinforced,  laminated,  supported or  similarly  combined with  other  materials,  n.e.c. in  chapter  39,level\_5:  470411 - Wood  pulp; chemicalwood pulp,  sulphite, (other  than dissolving  grades),  unbleached, of  coniferous  wood,level\_6:  95069925 -  Ice-hockey  and  field-hockey  articles and  equipment  (o/than balls  and skates),  and parts  ,country:  Canada,hts\_co  de:  9006610020,p  age: 1 | "status": true, "message": "Data  retrieved successfully", "data": {  "recordset": [ { "id": 1, "level\_1":  "Apparel & Textiles", "level\_2":  "Apparel", "level\_3": "Chapter 50 -Silk, including yarns and woven fabricthereof", "level\_4": "5005 - Yarn spun from silk waste, not put up for retail sale", "level\_5": "500500 - Silk; yarn spun from silk waste, not put up for  retail sale", "level\_6": "50050000 -Yarn spun from silk waste, not put up for retail sale","description": "Yrn  spun slk wste nt rtl sale gt 85% wgt slk/w", "hts\_code": "5005000010","country": "China", "updated\_date":  null, "value": 1 } ], "pagination": {  "total\_records": 43113, "page\_size":  10, "page\_no": 1 } } | API for  searching data |

1. **Template Project**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/template | Request  Type : Post | req.body | {"sku":707383,  "projectId":1328} | "status": true,  "message": "Data  created successfully",  "data": [ {  "newProjectId": 3672,  "name": "12/160SHEET  WHIT SAS  TOWE\_TURKMENISTAN\_  TURKMENISTAN",  "country":  "TURKMENISTAN" }, {  "elementId": 2644 }, {  "elementId": 2645 }, {  "subsectionId": 8594 }, {  "subsectionId": 8595 }, {  "subsectionId": 8596 }, {  "subsectionId": 8598 }, {  "subsectionId": 8597 }, {  "subsectionId": 8599 } ] | API for  creating  project from  template |

1. **Project Summary & Simplified - API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/chartCostSummar  y | Request  Type : Post | req.body | {project\_id:3971} | "status": true, "message": "Data  retrieved successfully", "data": {  "percentGap": 0, "POCost": 3.84,  "shouldCost": "35.12",  "chartCostData": [ { "name":  "Raw Material + Scrap",  "section\_id": 1, "value": 24 }, {  "name": "Overhead & Margin",  "section\_id": 4, "value": 1.11 }, {  "name": "Logistics", "section\_id":  5, "value": 1 }, { "name": "Duty /  Tariff", "section\_id": 6, "value":  9.01 }, { "name": "packaging",  "section\_id": 3, "value": 0 } ] } | API for  getting  Project  summary  chart datav |
| /api/v1/simplifiedProject | Request  Type : Post | req.body | {"project\_id":393  6,"skuList":[2825  43,282542],"perce  ntGap":"177.2","p  ercentBySection":  [{"name":"Raw  Material +  Scrap","value":"7  .1"},{"name":"Con  version","value":  "30.4"},{"name":"  Overhead &  Margin","value":"  1.0"},{"name":"Lo  gistics","value":  "37.5"},{"name":"  Duty /  Tariff","value":"  24.1"}],"ops":"po  sitive"} | "status": true, "message": "Data  created successfully" | API for  creating  simplified  project |
| api/v1/dashboardTableau | Request  Type : Post | NA | NA | "status": true, "message": "Data  retrieved successfully", "data": [ { "id": 1,  "display\_name": "Commodity  Intelligence", "description": null, "link":"\r\nhttp://tabdev.alixpartners.com/#/site/Tercio/views/ShouldCostAnalysisDemo/Commodities?:iid=4\r\n","img\_url":"\r\ncommodityicon.svg\r\n","status":1 },{ "id":2,"display\_name":"ShouldCostSummary","description":null,"link":"\r\nhttps://ip.alixpartners.com/#/site/Danke/views/ShouldCostAnalysisSimplified/ShouldCostAnalysis\r\n","img\_url":"\r\ncost-summary-icon.png", "status":1}] | API for getting  Dashboard  tableau data |

1. **Conversion Final Formula – API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

1. **Element Delete– API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v2/deleteElement | Request  Type : Post | req.body | {"element\_id":560} | "status": true,  "message": "Data deleted  successfully" | For  deleting  Added  Materials. |

1. **Template Master – API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v2/template/master | Request  Type : Post | req.body | { "proj\_id":null,  "country":"INDIA",  "category\_group":"  House","category":"PAPR","sub\_category":"TOWEL",  "status":"enable",  "description":"abc"} | "status": true,  "message": "Data created  successfully" | For  creating  templates  on template  library |

1. **Template Delete - API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v2/template/delete | Request  Type : Post | req.body | { "id":12 } | "status": true,  "message": "Data deleted  successfully" | For  deleting  Specific  template  from  template  library |

1. **Template Preview - API**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| /api/v1/chartCostSummar  y | Request  Type : Post | req.body | {project\_id:3971} | "status": true,  "message": "Data  retrieved  successfully", "data": {  "percentGap": 0,  "POCost": 3.84,  "shouldCost": "35.12",  "chartCostData": [ {  "name": "Raw  Material + Scrap",  "section\_id": 1,  "value": 24 }, {  "name": "Overhead &  Margin", "section\_id":  4, "value": 1.11 }, {  "name": "Logistics",  "section\_id": 5,  "value": 1 }, { "name":  "Duty / Tariff",  "section\_id": 6,  "value": 9.01 }, {  "name": "packaging",  "section\_id": 3,  "value": 0 } ] } | show  preview in  template  library |
| api/v1/projectSummaryTable | Request  Type : Post | req.body | {"project\_id":34662} | "status": true,  "message": "Data deleted  successfully" | show  preview in  template  library |

1. **Mass Upload - API’s**

**Import:**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| api/v2/template/create/data | Request  Type : Post | req.body | mass upload  template excel  data in json format | "status": true,  "message": "Data created  successfully" | data upload  for complex  project in  mass  upload |

**Calculate Should Cost:**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| api/v2/template/create | Request  Type : Post | req.body | { "id":21 } | "status": true,  "message": "Template started  creating"  "data": {"projectId": 2103} | For  creating  Specific  template  or project  from mass  upload |

**Delete:**

| **API endpoint** | **Request type** | **Input** | **Payload** | **API response** | **Description** |
| --- | --- | --- | --- | --- | --- |
| api/v2/template/delete/Data | Request  Type : Post | req.body | { "id":15,  "projectId": null } | "status": true,  "message": "Data deleted  successfully" | For  deleting  Specific  row  from mass  upload |

# Reporting Solution (Low Level Design)

(Tableau)

* 1. **Semantic Layer Design**

Please describe details on semantic layer

* 1. **Reporting Layouts / Design**

Please provide details for each report in the below suggested format.

| <Report ID> | <Report Name> |
| --- | --- |
| Header of Report | <Provide header of Report |
| **Parameters** |  |
| <Parameter1> | Details (Prompt, drop down, list) |
| <Parameter2> | Details (Prompt, drop down, list) |
| **Columns** |  |
| <Columne1> | Format (Currency, description, rate) |
| <Columne2> | Format (Currency, description, rate) |
| **Filter** |  |
| <Filter1> | Where clause |
| <Filter2> | Where clause |

Please add all applicable properties for the project. (e.g. sub-totals)

# 

# UI Solution (Low Level Design)

# 

# Page Process/Navigation Flow

* + 1. **User Interaction Flow**

Below are the details of all the pages of the application.

**Build should cost:** All the products (SKUs) for which the should cost can be calculated are listed here. It shows the status as “Started” which means a should cost project is started for that SKU and “Not started” meaning no project has been created for that SKU.

Graphical user interface, table

Description automatically generated

**Modify should cost:** All the should cost projects are listed here. We can filter the projects based on different parameters. Some projects are marked as final based on the predefined logic. You can also mark or unmark a project as final. The toggle button available shows all the projects which are marked as final when turned on.

Graphical user interface, table

Description automatically generated

# Conversion library: The conversion library has the data about all the conversion rates, currencies, machine conversion rates, labor, logistics, etc.

A screenshot of a computer

Description automatically generated

# Machine library: It Has the data about all the machines used for manufacturing, packaging, etc. of a material.

Graphical user interface, table

Description automatically generated

**Template library:** Should cost can also be calculated using a premade template which has all the required data. Here you can view all the available templates. Also, when a project is created from scratch, you can mark it as a template to use the same details for creating other projects. These marked templates can also be viewed in the template library.

Graphical user interface, table

Description automatically generated

**Project summary:** The summary of the project has two parts. One is a waterfall graph with all the costs for the projects and a summary table which has the similar cost breakdown.

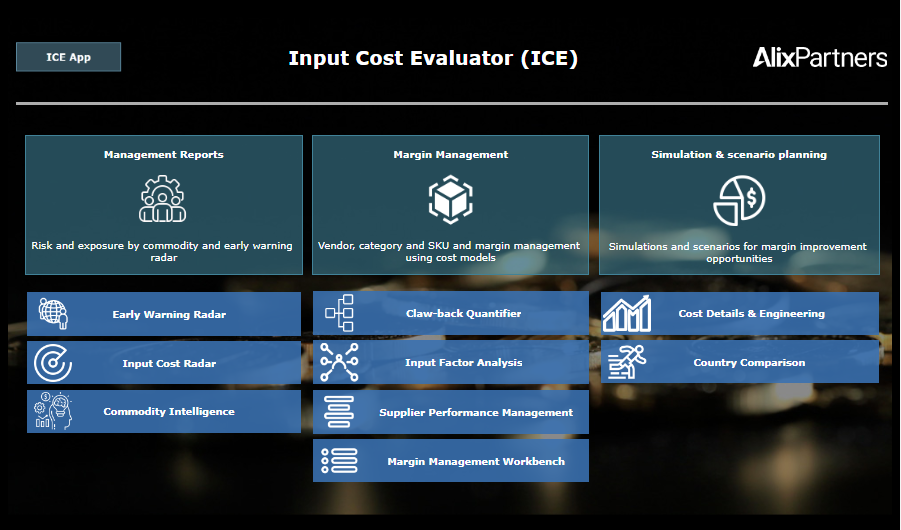
Chart, waterfall chart

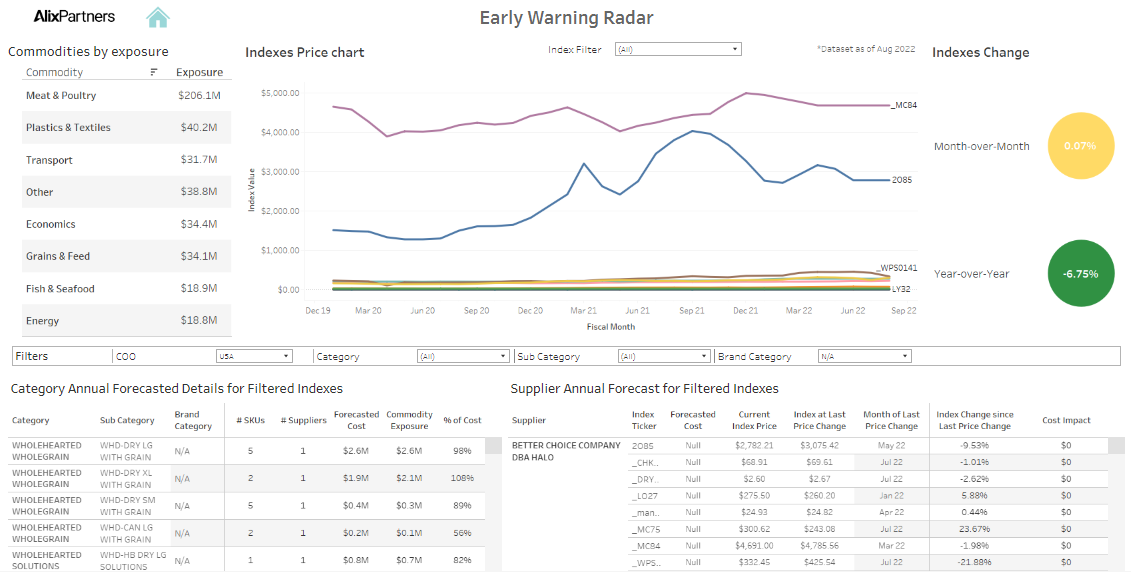
Description automatically generated

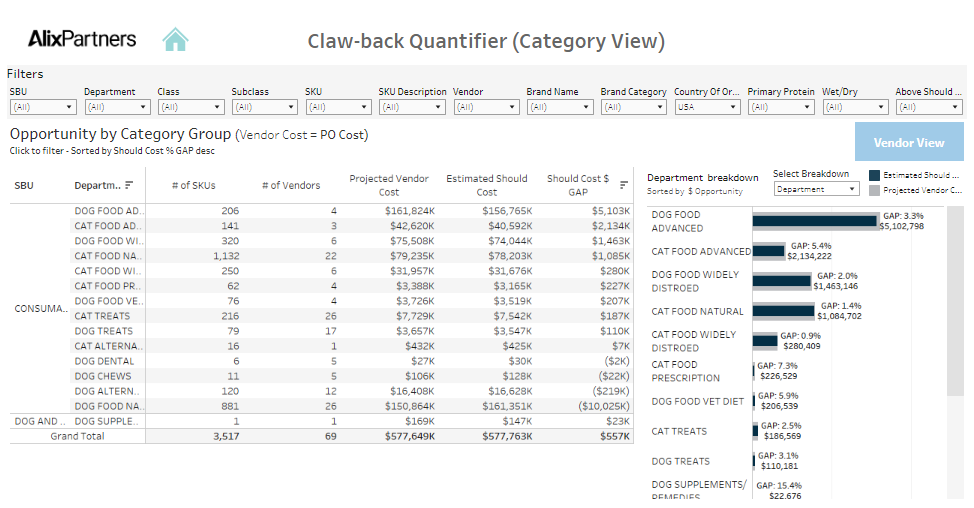
Graphical user interface, table

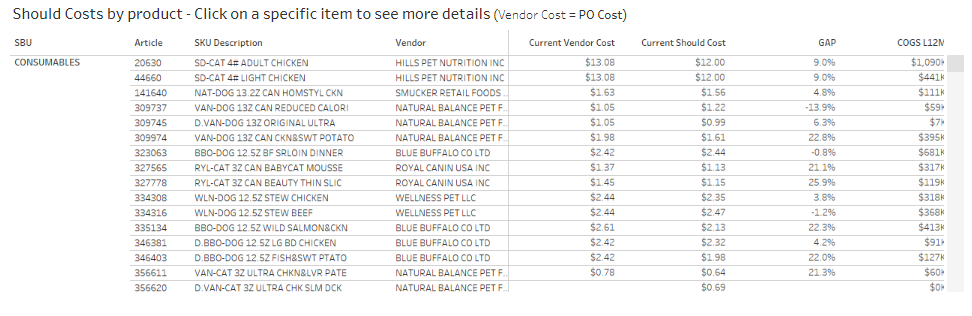
Description automatically generated

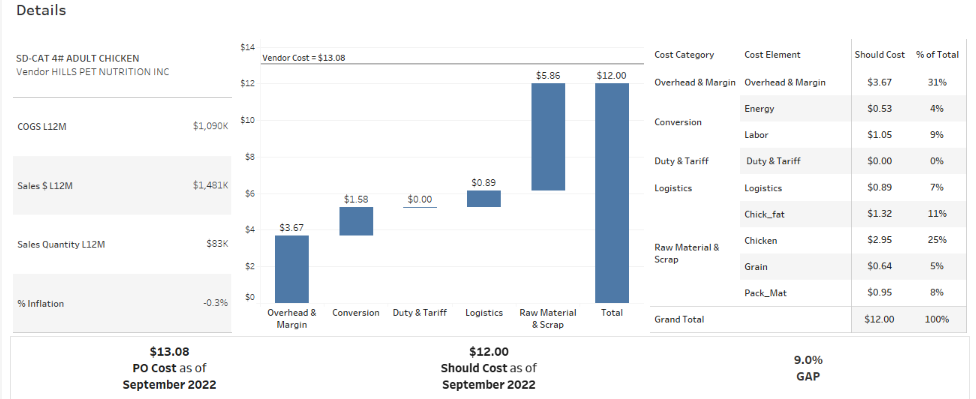
* + 1. **Reports (Graphs/ Charts/Tables/Filters/Downloads).**

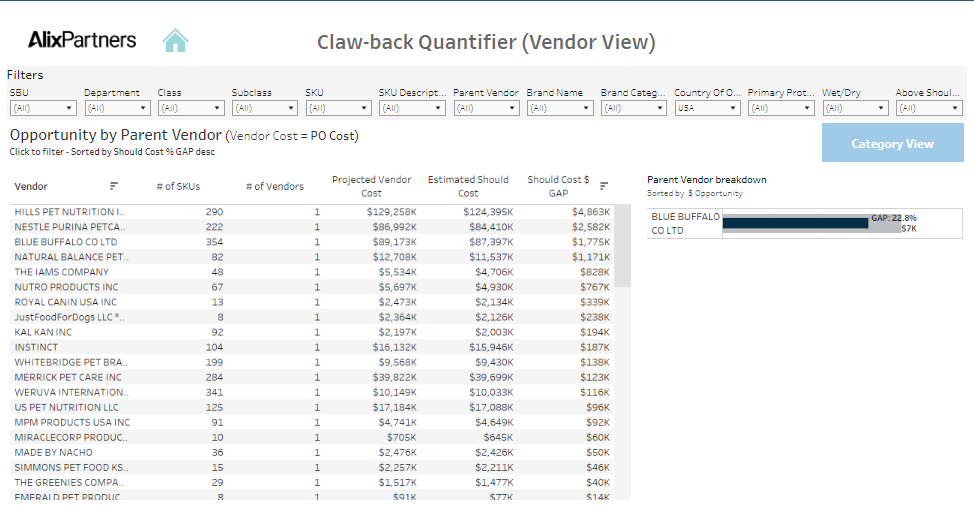
Landing Page: 

Early Warning Radar: 

Claw-Back Quantifier: 



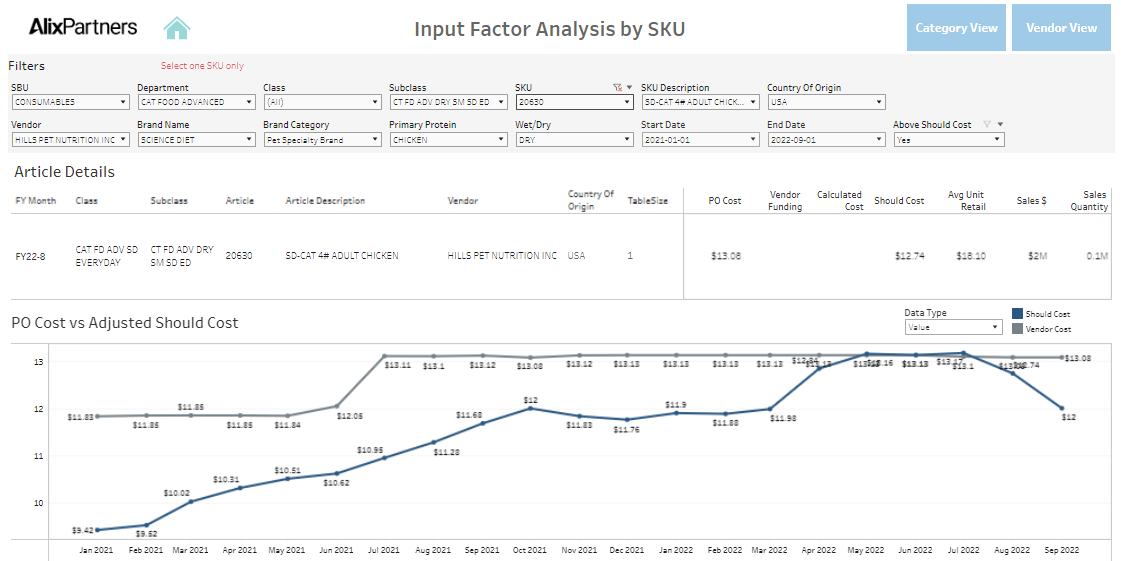


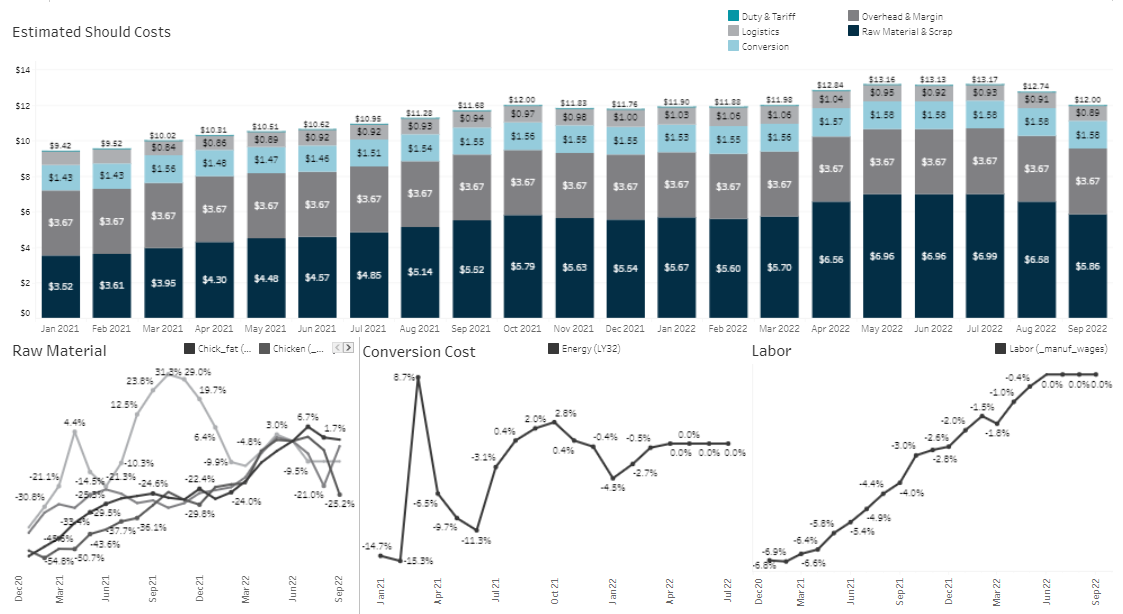


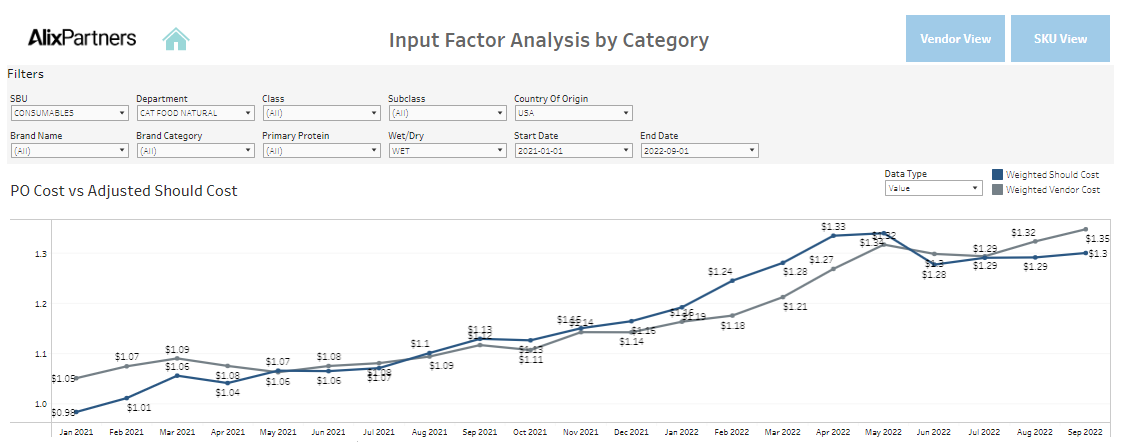
Country Comparison:

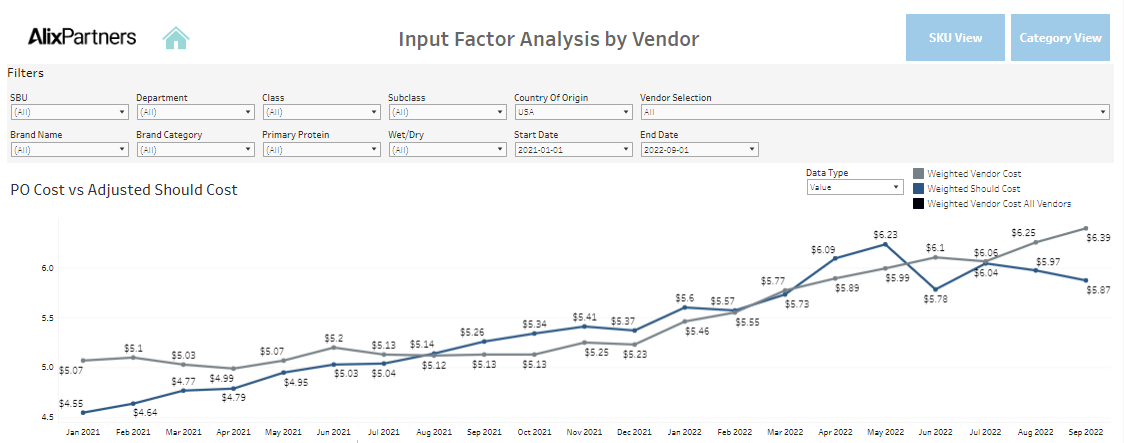


Input Factor Analysis:

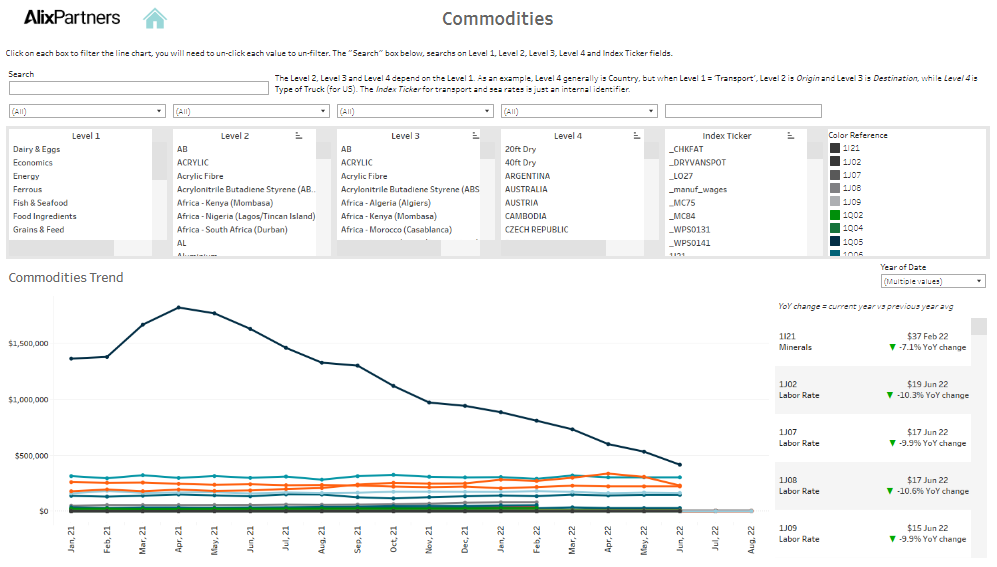








Commodities:





* + 1. **Search**   
       There is a single search box present on all the pages. It can search on one of the following columns as mentioned below:

1. Build should cost - SKU
2. Modify should cost - project\_code
3. Conversion library - ID
4. Template library - ID
5. Machine library - ID

Apart from the search box, there are filters available on all the pages for the important parameters. One can select the item from the drop-down list and can also search for the required item in the search box inside the filter.

# 

# Backend (Low Level Design) (Scripting/SQL Processing)

* **SQL Triggers**

1. **Project Code:** Triggers are created to generate a project code for a newly created should cost project. The project code also needs to be updated whenever the country of a project is changed.
2. **Final project flag:** The final project trigger marks a project as final since the user can create multiple projects for the same country and product, one project from that group needs to be marked as final. When the user marks another project from the same group as final, the previous project gets unmarked. Hence, there can only be one final project.

* **SQL Views**

Following are the different views created to use them as different data sources in Tableau dashboard. Various joins, calculations and some aggregations done in these views.

1. Icr.basket\_movers
2. Viz.all\_commodities\_data
3. Icr.basket\_over\_time
4. Icr.commodities\_chart
5. Viz.commodities\_data
6. Viz.early\_warning\_radar
7. Viz.input\_factor\_analysis\_dash
8. Viz.should\_cost\_analysis\_dash
9. viz.vendor\_cost\_structure

# References

Please provide references like web sites / presentations / design best practices / customer related websites.

# Appendix

Please mention any more information about customer emails / notes / screenshots.