

**HIGH LEVEL DESIGN (HLD) SIGN-OFF**

**Authorization Memorandum**

This form is to sign-off completion of the Design Phase for **CAMPUS MANAGEMENT SOFTWARE AT WORLD SKILL CENTER (WSC).**

World Skill Centre (WSC) acknowledges receipt of the deliverables as part of the Design Phase through the submission of this document.

|  |  |
| --- | --- |
| **MODULE NAME** | **SOE : Training Of Trainers(ToT) module** |
| **MODULE OWNER** | **Mr. Subhasis Das** |

**WSC AUTHORITY NAME AND SIGNATURE**

**SOUL AUTHORITY NAME AND SIGNATURE**

Logo

Description automatically generated Logo

Description automatically generated

Logo

Description automatically generated

**High Level Design (HLD)**

Of

**Training Of Trainers (ToT)**

For Implementation of

**Campus Management Software**

at

**World Skill Center (WSC)**

**Department - School Of Engineering (SOE)**

**Sustainable Outreach And Universal**

**Leadership (SOUL) Limited**

**Confidentiality**

This document is a strictly confidential communication to and solely for the use of the recipient and may not be reproduced or circulated without SOUL Ltd’s prior written consent. If you are not the intended recipient, you may not disclose or use the information in this documentation in any way

**Document Control History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Version No.** | **Created By** | **Affected Sections** | **Reviewed By** | **Updated By** |
| 05-Aug-2023 | 1.0 | Sayantan Ghosh | All |  |  |
| 05-Sep-2023 | 1.1 |  | Section 1,10, project control, 5, prototype | Sharmistha Panda |  |

# Project Control

|  |  |
| --- | --- |
| **Package Number:** | PKG 0111 |
| **Project Name:** | Campus Management Software At World Skill Center (WSC) |
| **Location:** | Bhubaneswar |
| **Customer Name:** | World Skill Center (WSC) |
| **Customer Contact Information:** | Mancheswar, Bhubaneswar |
| **Project Manager / Email ID:** | Sharmistha Panda  sharmistha.panda@soulunileaders.com |
| **Module SME / Email ID:** | Subhasis Das  subhasis.das@worldskillcenter.org |

Table Of Contents

[HIGH LEVEL DESIGN (HLD) SIGN-OFF 1](#_Toc14021)

[Authorization Memorandum 1](#_Toc10082)

[Project Control 3](#_Toc30160)

[1 Introduction 7](#_Toc21479)

[1.1 Background 7](#_Toc7918)

[1.2 Scope and Purpose of the document 7](#_Toc2471)

[1.3 Assumptions 8](#_Toc13057)

[1.4 Dependencies 8](#_Toc21396)

[1.5 Current IT Environment 8](#_Toc20582)

[1.6 Document Structure 8](#_Toc24837)

[1.7 Required System Architecture 8](#_Toc29143)

[1.8 Interoperability Framework 9](#_Toc12160)

[1.9 Technical Decision Summary 10](#_Toc15767)

[1.10 Architectural Risks & Mitigation 10](#_Toc14284)

[1.11 Reference 11](#_Toc11957)

[2 Enterprise Architecture Framework (Business Architecture) 11](#_Toc24720)

[2.1 Business Drivers 11](#_Toc29096)

[2.2 Business Concerns 11](#_Toc19223)

[2.3 Business Goals 11](#_Toc22166)

[2.4 Business Value Chain 12](#_Toc29414)

[2.5 Business Context Diagram 12](#_Toc3839)

[2.6 Business Process Flow 13](#_Toc8006)

[2.6.1 Training of Trainers 13](#_Toc22955)

[2.6.2 System Context Diagram 13](#_Toc24059)

[2.7 Conceptual Architecture 13](#_Toc5892)

[3 Architecture Views 14](#_Toc14511)

[3.1 Trainer 14](#_Toc20805)

[3.1.1 Use Case Diagram 14](#_Toc8941)

[3.1.2 Design Workflow 15](#_Toc24360)

[3.1.3 Validations 15](#_Toc19601)

[3.1.4 Field List: 15](#_Toc15606)

[3.1.5 Processes After Form Submission 16](#_Toc9498)

[3.1.6 Process Flow: 16](#_Toc22311)

[3.1.7 Pre-requisites and Dependency 17](#_Toc32329)

[3.2 ToT Participant 17](#_Toc6899)

[3.2.1 Use Case Diagram 17](#_Toc28909)

[3.2.2 Design Workflow 17](#_Toc1161)

[3.2.3 Validations 17](#_Toc31945)

[3.2.4 Field List: 18](#_Toc24509)

[3.2.5 Processes After Form Submission 20](#_Toc25848)

[3.2.6 Process Flow: 20](#_Toc22383)

[3.2.7 Pre-requisites and Dependency 20](#_Toc5189)

[3.3 ToT Participant Selection 20](#_Toc4353)

[3.3.1 Use Case Diagram 20](#_Toc28995)

[3.3.2 Design Workflow 21](#_Toc15500)

[3.3.3 Validations 21](#_Toc2095)

[3.3.4 Field List: 22](#_Toc1980)

[3.3.5 Processes After Form Submission 22](#_Toc24804)

[3.3.6 Process Flow: 22](#_Toc4210)

[3.3.7 Pre-requisites and Dependency 23](#_Toc18790)

[3.4 ToT Participant Enrollment 23](#_Toc7586)

[3.4.1 Use Case Diagram 23](#_Toc25040)

[3.4.2 Design Workflow 24](#_Toc22543)

[3.4.3 Validations 24](#_Toc23723)

[3.4.4 Field List: 25](#_Toc18881)

[3.4.5 Processes After Form Submission 25](#_Toc16113)

[3.4.6 Process Flow: 26](#_Toc25207)

[3.4.7 Pre-requisites and Dependency 26](#_Toc1364)

[3.5 Participant Group 26](#_Toc28770)

[3.5.1 Use Case Diagram 26](#_Toc121)

[3.5.2 Design Workflow 27](#_Toc14308)

[3.5.3 Validations 27](#_Toc2116)

[3.5.4 Field List: 28](#_Toc27184)

[3.5.5 Processes After Form Submission 29](#_Toc26229)

[3.5.6 Process Flow: 29](#_Toc31226)

[3.5.7 Pre-requisites and Dependency 29](#_Toc6938)

[3.6 ToT Class Schedule 29](#_Toc1692)

[3.6.1 Use Case Diagram 29](#_Toc29093)

[3.6.2 Design Workflow 30](#_Toc6480)

[3.6.3 Validations 30](#_Toc21417)

[3.6.4 Field List: 30](#_Toc3455)

[3.6.5 Processes After Form Submission 31](#_Toc28299)

[3.6.6 Process Flow: 31](#_Toc1069)

[3.6.7 Pre-requisites and Dependency 32](#_Toc11177)

[3.7 ToT Participant Attendance 32](#_Toc1025)

[3.7.1 Use Case Diagram 32](#_Toc14018)

[3.7.2 Design Workflow 33](#_Toc12943)

[3.7.3 Validations 33](#_Toc27819)

[3.7.4 Field List: 34](#_Toc14523)

[3.7.5 Processes After Form Submission 34](#_Toc7313)

[3.7.6 Process Flow: 34](#_Toc13170)

[3.7.7 Pre-requisites and Dependency 35](#_Toc6227)

[3.8 Participant Attendance Tool 35](#_Toc23320)

[3.8.1 Use Case Diagram 35](#_Toc13504)

[3.8.2 Design Workflow 36](#_Toc8014)

[3.8.3 Validations 36](#_Toc8385)

[3.8.4 Field List: 36](#_Toc8705)

[3.8.5 Processes After Form Submission 37](#_Toc29354)

[3.8.6 Process Flow: 37](#_Toc5090)

[3.8.7 Pre-requisites and Dependency 37](#_Toc1360)

[3.9 Assignment 38](#_Toc17849)

[3.9.1 Use Case Diagram 38](#_Toc19092)

[3.9.2 Design Workflow 38](#_Toc7472)

[3.9.3 Validations 38](#_Toc10115)

[3.9.4 Field List: 39](#_Toc20741)

[3.9.5 Processes After Form Submission 39](#_Toc20029)

[3.9.6 Process Flow: 39](#_Toc12233)

[3.9.7 Pre-requisites and Dependency 40](#_Toc27708)

[3.10 Assignment Upload 40](#_Toc3473)

[3.10.1 Use Case Diagram 40](#_Toc30646)

[3.10.2 Design Workflow 41](#_Toc6218)

[3.10.3 Validations 41](#_Toc32047)

[3.10.4 Field List: 41](#_Toc3983)

[3.10.5 Processes After Form Submission 42](#_Toc5748)

[3.10.6 Process Flow: 42](#_Toc1063)

[3.10.7 Pre-requisites and Dependency 42](#_Toc16023)

[3.11 Assignment Declaration 43](#_Toc9377)

[3.11.1 Use Case Diagram 43](#_Toc4772)

[3.11.2 Design Workflow 43](#_Toc1959)

[3.11.3 Validations 43](#_Toc19751)

[3.11.4 Field List: 44](#_Toc10745)

[3.11.5 Processes After Form Submission 45](#_Toc31422)

[3.11.6 Process Flow: 45](#_Toc17712)

[3.11.7 Pre-requisites and Dependency 45](#_Toc6212)

[3.12 Assignment Evaluation 45](#_Toc3518)

[3.12.1 Use Case Diagram 46](#_Toc3933)

[3.12.2 Design Workflow 46](#_Toc6597)

[3.12.3 Validations 46](#_Toc27761)

[3.12.4 Field List: 47](#_Toc32690)

[3.12.5 Processes After Form Submission 47](#_Toc16718)

[3.12.6 Process Flow: 47](#_Toc29389)

[3.12.7 Pre-requisites and Dependency 48](#_Toc523)

[3.13 Assignment Evaluation Tool 48](#_Toc5395)

[3.13.1 Use Case Diagram 48](#_Toc21991)

[3.13.2 Design Workflow 48](#_Toc12566)

[3.13.3 Validations 49](#_Toc25654)

[3.13.4 Field List: 49](#_Toc25535)

[3.13.5 Processes After Form Submission 50](#_Toc9626)

[3.13.6 Process Flow: 50](#_Toc15641)

[3.13.7 Pre-requisites and Dependency 51](#_Toc17431)

[3.14 Final Assignment Result 51](#_Toc22683)

[3.14.1 Use Case Diagram 51](#_Toc10442)

[3.14.2 Design Workflow 51](#_Toc12416)

[3.14.3 Validations 51](#_Toc3140)

[3.14.4 Field List: 52](#_Toc19764)

[3.14.5 Processes After Form Submission 53](#_Toc18140)

[3.14.6 Process Flow: 53](#_Toc14565)

[3.14.7 Pre-requisites and Dependency 53](#_Toc18552)

[3.15 Final Result Declaration Tool 53](#_Toc8359)

[3.15.1 Use Case Diagram 53](#_Toc7058)

[3.15.2 Design Workflow 54](#_Toc6109)

[3.15.3 Validations 54](#_Toc13708)

[3.15.4 Field List: 55](#_Toc22463)

[3.15.5 Processes After Form Submission 55](#_Toc9876)

[3.15.6 Process Flow: 55](#_Toc25653)

[3.15.7 Pre-requisites and Dependency 56](#_Toc3490)

[4 Tools And Technologies 56](#_Toc2142)

[5 Non-Functional Requirements 56](#_Toc28795)

[5.1 Performance 56](#_Toc25764)

[5.2 Reliability 56](#_Toc5150)

[5.3 Usability 56](#_Toc14084)

[5.4 Compatibility 57](#_Toc31424)

[5.5 Security 57](#_Toc10477)

[5.6 Compliance 57](#_Toc30263)

[5.7 Documentation 57](#_Toc31513)

[5.8 Security: User access and authorization 57](#_Toc2461)

[Prototype 61](#_Toc7237)

[6 Prototype 62](#_Toc10404)

[6.1 Trainer 63](#_Toc31155)

[Figure 1 Trainer 63](#_Toc6772)

[6.2 ToT Participant 63](#_Toc22803)

[Figure 2 ToT Participant 64](#_Toc25729)

[6.3 ToT Participant Selection 64](#_Toc11910)

[6.4 ToT Participant Enrollment 64](#_Toc1272)

[6.5 Participant Group 65](#_Toc14286)

[6.6 ToT Class Schedule 65](#_Toc16824)

[6.7 ToT Participant Attendance 66](#_Toc11755)

[6.8 Participant Attendance Tool 66](#_Toc14320)

[6.9 Assignment 67](#_Toc10421)

[6.10 Assignment Upload 67](#_Toc31196)

[6.11 Assignment Declaration 68](#_Toc27699)

[6.12 Assignment Evaluation 69](#_Toc13112)

[6.13 Assignment Evaluation Tool 69](#_Toc21952)

[6.14 Final Assignment Result 70](#_Toc21077)

[6.15 Final Result Declaration Tool 70](#_Toc3899)

[7 Definitions and Acronyms 71](#_Toc25352)

[8 Deployment Description 71](#_Toc8406)

# Introduction

The World Skill Center (WSC) is a premier advanced skill training institute established by the Government of Odisha, Skill Development and Technical Education Department, through the Odisha Skill Development Authority (OSDA). The WSC will impart advanced skill training in eight trades from engineering and service sectors. WSC caters primarily to induct the best talent from ITIs and Polytechnics and training them to become globally employable in emerging areas such as "Industry 4.0". WSC is housed in a state-of-the-art, 18-storey, air-conditioned building with nearly half a million square feet of space in the heart of capital city of Bhubaneswar.

With the implementation of campus management software for WSC the goal is to streamline the operations and functions of the campus by integrating various processes, such as admissions, course registration, academic progress tracking, and financial management, HRMS, Procurement and Inventory management, etc into a unified system. The implementation also aims at providing a user-friendly interface for all stakeholders, making it easier for them to access the necessary information and complete their tasks with ease. The modules to be covered during the implementation of the software include:

|  |  |
| --- | --- |
| Students Management Modules | Infrastructure Management |
| Academic System | Finance and Accounting System |
| Procurement & Inventory Management | Training and Placement |
| Human Resources Management System | Application Integration |

# 

## Background

The Web Based Campus Management Application at World Skill Centre (WSC) application is required by WSC for the smooth operation of all departments / support functions with on-line delivery of services to all stakeholders.

The project aims to create a mechanism to provide the basis for evolution of an IT enabled state of the art workflow automation system in a planned manner.

## Scope and Purpose of the document

The design documents track the necessary information required to effectively define architecture and system design in order to give the development team guidance on the architecture of the system to be developed. Its intended audience is the project manager, project team, and development team. Some portions of this document, such as the user interface (UI), may be shared with the client/user, and other stakeholders whose input/approval into the UI is needed.

This document covers all the design aspects of Training module. This module helps in organizing the entire training setup such as Participants Database, Application forms, Participant attendance etc.

The scope of Training of Trainers module

* Organization of Training of Trainer events
* Participant record creation
* Participant selection
* Assignment allocation and evaluation
* Recording attendance
* Certificate publishing

## Assumptions

The assumptions are listed as follows:

* The required hardware and software requirements will be provided
* Data loading to be handled by WSC

All external entities will provide their interface for application integration

## Dependencies

The ERP system will be dependant on external interfaces for integration services. Following are few external interfaces:

* Payment Gateway
* Communication channels such as – Email, SMS, WhatsApp
* Microsoft 365
* Biometric System
* Website

## Current IT Environment

|  |  |
| --- | --- |
| **Hardware Requirements** | **Software Requirements** |
| Processor: Intel CPU with at least 8 cores | Operating System: Ubuntu 22.04 LTS |
| RAM: 16 GB to 32 GB | Web Server: Nginx |
| Storage: A minimum of 160 GB SSD | Database server: MariaDB |
| Network: 1 Gbps | Python: 3.10 or later |

## Document Structure

The following sections are part of this document :

* Business goals, objectives and requirements
* Understand business context and interactions
* Conceptual design
* Overall system context
* Understand use-case and scenarios
* Define implementation

## Required System Architecture

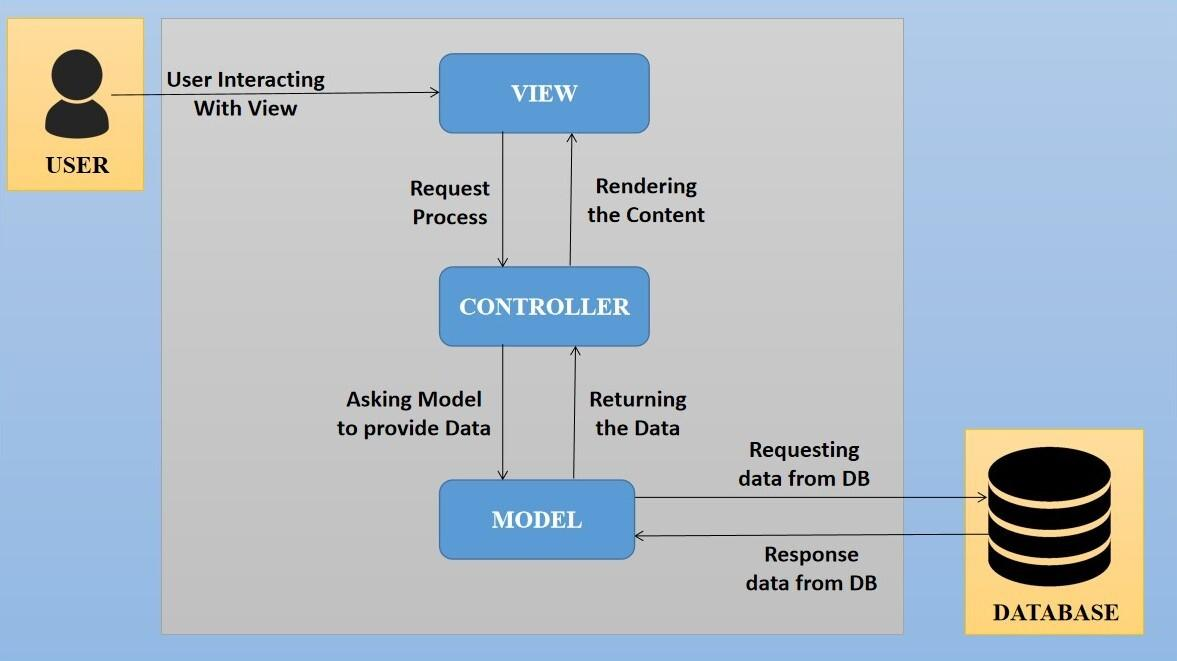


Figure 1 System Architecture

The proposed design is an architectural pattern that separates an application into three main logical

components: The **Model**, the **View** and the **Controller**. Each of these components are built to handle

specific development aspects of the application like:

* The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data
* The View component is used for all the UI logic of the application. It generates a user interface for the user
* The controller is the component that enables the interconnection between the views and the model so it acts as an intermediary

## Interoperability Framework

Frappe is a full stack, batteries-included, web framework written in Python and JavaScript with MariaDB as the database. It is pretty generic and can be used to build database-driven apps with an elegant and consistent UI.



Figure 2 Framework

## Technical Decision Summary



Figure 3 Technical Requirements

## Architectural Risks & Mitigation

|  |  |
| --- | --- |
| **Risks** | **Recommendation** |
| Inability to provide the required hardware resources (Server) for installation of the ERP product | The issue should be brought up during cadence meetings, and the OCAC/WSC IT team should assure timely availability of the required server |
| Unavailability of API for third party integration | The project timeline schedule needs to be shared with third parties to avoid the delay in integration. Also, inclusion of external teams to meet timelines |
| The number of users accessing the application exceeds its limit (as specified in the proposal document), leading to performance issues | WSC leadership team to ensure the scalability of the hardware / software resources |
| End Users of the application unavailable for the training to be provided by SOUL | SOUL to inform WSC Stakeholders on training plan in advance and engage early for training |

## Reference

|  |  |
| --- | --- |
| ***Sl No.*** | ***Document Name*** |
| 1 | SRS for Training of Trainers |
|  |  |

# Enterprise Architecture Framework (Business Architecture)

## Business Drivers

The following points motivate the business efficacy of Campus Management Software at World Skill Center (WSC) system:

* Integrate the various functions such as Admission, Academics, Examination, Training & Placement, HRMS, Procurement, Finance into a single platform which can be customized as per user requirements
* Implement a common framework for the system and sub systems
* Provide a web based application for the users to interact with the system with Role based Access and consistent look and feel

## Business Concerns

Campus Management Software At World Skill Center application is designed to address the following concerns:

Student Management: To help the students / trainers manage the students admission activities, starting from initial communication to course enrollment

Financial Management: Managing finances is a fundamental concern for businesses. This includes budgeting, cash flow management, ledger maintenance, balance sheet,etc

Employee Engagement: Attracting, retaining, and engaging skilled and motivated employees is critical. Human resources concerns also include training, performance management

Procurement management: The source-to-settle process. It encompasses the evaluation, selection, and creation of formal contractual agreements as well as managing the company's ongoing supplier relationships

## Business Goals

The goal is to implement web based Campus Management Application System for efficient internal functioning of World Skill Center (WSC) with on-line delivery of services to each stakeholder of WSC supported by a suitable, robust, secure and reliable system

## Business Value Chain

The ERP system's business value chain typically encompasses the following key components:

Procurement Management: This includes the processes related to sourcing and acquiring raw materials, goods, or services required for the organization's operations. The ERP system streamlines procurement activities, such as supplier management, purchase requisitions, purchase orders, and inventory management, leading to cost savings and better supply chain management.

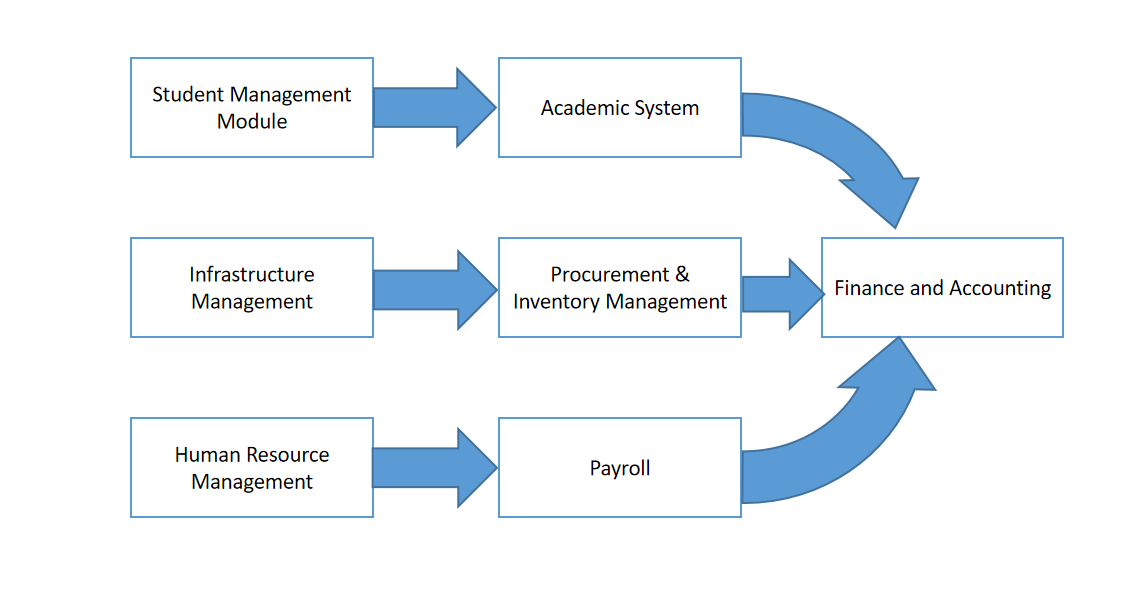
Production/Operations Management: The ERP system facilitates the planning, scheduling, and execution of production processes. It optimizes resource allocation, tracks work progress, manages bills of materials, and monitors production costs to enhance overall operational efficiency and quality.

Inventory Management: The ERP system enables real-time tracking of inventory levels, stock movements, and stock outs. It helps in maintaining optimal inventory levels, reducing carrying costs, and ensuring timely availability of products.

Financial Management: ERP systems centralize financial data and automate accounting, financial reporting, budgeting, and financial analysis. This streamlines financial processes, enhances accuracy, and provides management with a clear financial overview for better decision-making.

Human Resources Management: ERP systems handle various HR functions, including payroll, employee records, recruitment, performance management, and training. This leads to streamlined HR processes, improved workforce management, and enhanced employee satisfaction.

## Business Context Diagram



## Business Process Flow

### Training of Trainers

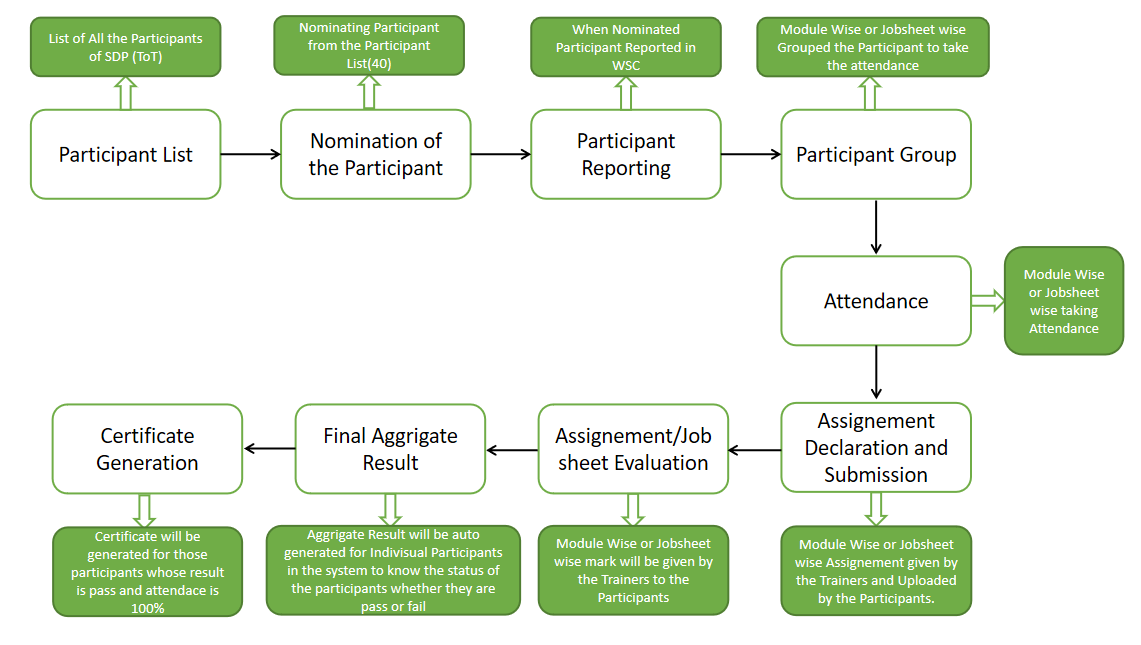
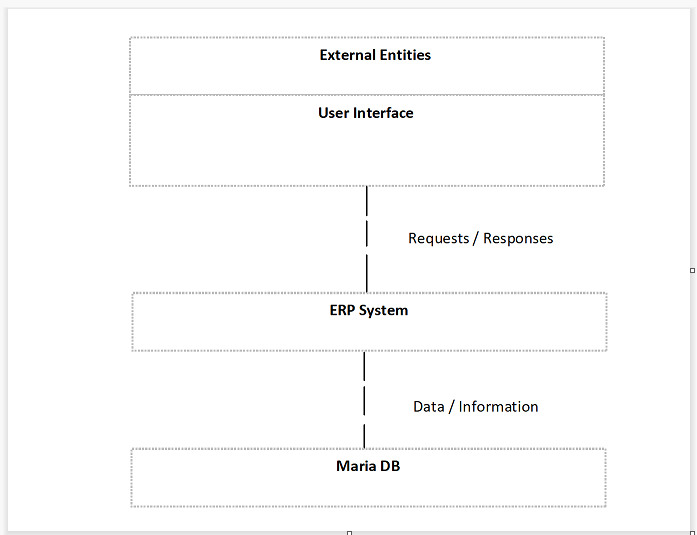


Figure 4:Training of Trainers Process Flow

### System Context Diagram



## Conceptual Architecture

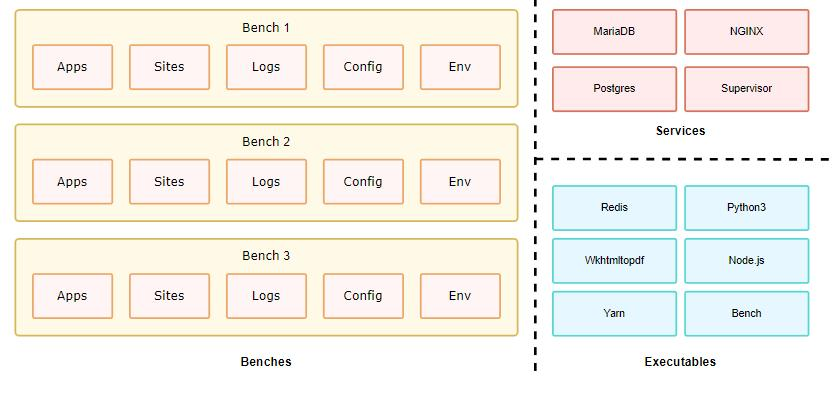


Figure 7 Multitenant Architecture

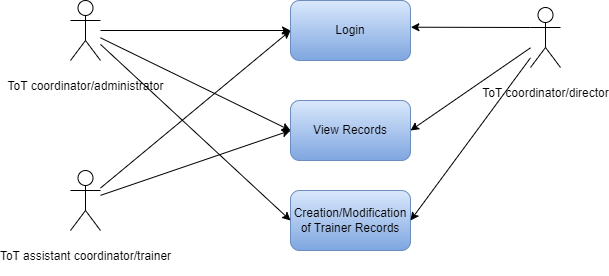
A single bench can host multiple combinations of sites and apps. For the most part, having a single bench works for hosting hundreds of sites that depend on the same versions of said app, given you've scaled up the workers. You can host multiple versions of the application on the same server by creating multiple benches parallelly. The following diagram hints to the system dependencies and how they are used.

# Architecture Views

## Trainer

The Trainer screen is to be used by the ToT officials for creating and modifying records of trainers for the ToT Programs.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Grading Scale, Get Participants
* Fields displayed on condition : Not Applicable
* Linked Fields : The Employee field is linked with the Employee screen.The Gender field is linked with the Gender screen.The Department field is linked with the Department screen.The Academic Year field is linked with the Academic Year screen.The Academic Term field is linked with the Academic Term screen.The Department field is linked with the Department screen.The Program field is linked with the Program screen.The Course field is linked with the Course screen.The Student Group field is linked with the Student Group screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Autoname(self) - This function when triggered decides whether to set the default naming series for a record or use the data from the instructor field for creating the naming series.
* Validate(self) - This function when triggered calls the validate \_duplicate\_employee() function
* Validate\_duplicate\_employee() - This function when triggered checks whether another record exists with the same details as the current record that is being created.
* Get\_timeline\_data(doctype\_name) - This function when triggered gets the timeline related parameted for the particular instructor.

**Notification:**

* Not Applicable

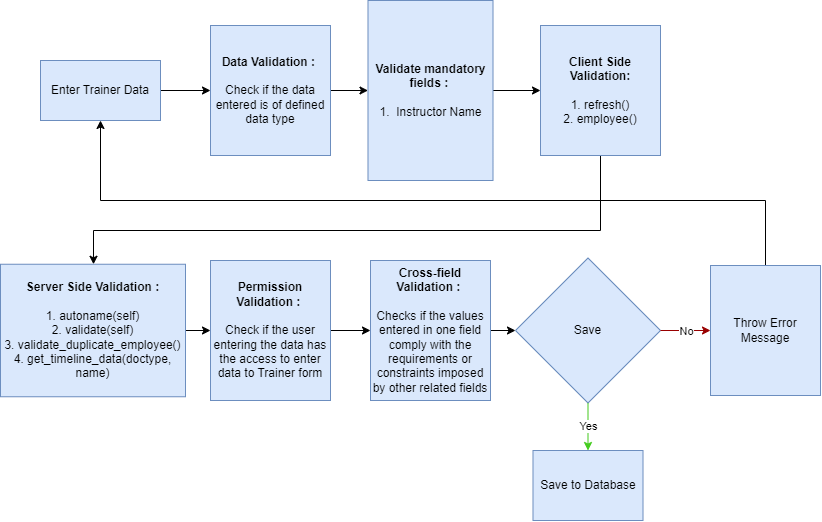
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Instructor Name | Data | Yes |  |
| 2 | Employee | Link |  | Employee |
| 3 | Gender | Link |  | Gender |
| 4 | Status | Select |  | Active  Left |
| 5 | Naming Series | Select |  | EDU-INS-.YYYY.- |
| 6 | Department | Link |  | Department |
| 7 | Image | Attach Image |  |  |
| 8 | Instructor Log | Table |  | Instructor Log |
| **Instructor Log** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 9 | Academic Year | Link | Yes | Academic Year |
| 10 | Academic Term | Link |  | Academic Term |
| 11 | Department | Link |  | Department |
| 12 | Program | Link | Yes | Program |
| 13 | Course | Link |  | Course |
| 14 | Student Group | Link |  | Student Group |
| 15 | Other details | Samll Text |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:



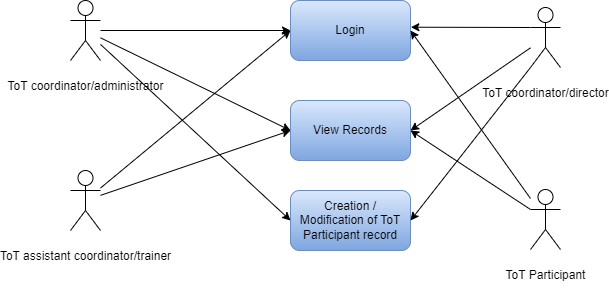
### Pre-requisites and Dependency

* Not Applicable

## ToT Participant

The ToT Participant screen is to be used by the ToT officials for creating and modifying records of participants who are eligible for a ToT program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : First Name, Last Name, User ID, HRMS ID, Participant ID, Designation, Department, Naming Series, Participant Email Address, Participant Mobile Number, Adhaar Number
* Fields displayed on condition : Not Applicable
* Linked Fields : The Participant ID field is linked with the Student screen.The Participant Caste field is linked with the Student Category screen.The Gender field is linked with the Gender screen.The State field is linked with the State screen.The District field is linked with the Districts screen.The Block field is linked with the Blocks screen.The Country field is linked with the Country screen.The Programs field is linked with the Programs screen.The Academic Year field is linked with the Academic Year screen.The Participant Selection field is linked with the ToT Participant Selection screen.The Participant Enrollment field is linked with the ToT Participant Enrollment.The Semester field is linked with the Program screen.The Academic Term field is linked with the Academic Term screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) - This function joins values in the First Name, Middle Name and Last Name fields to get and store the full name of the participant as a single string.
* Phone\_no\_validate(self) -This function when triggered validates the phone number entered in the doctype.

**Notification:**

* Not Applicable

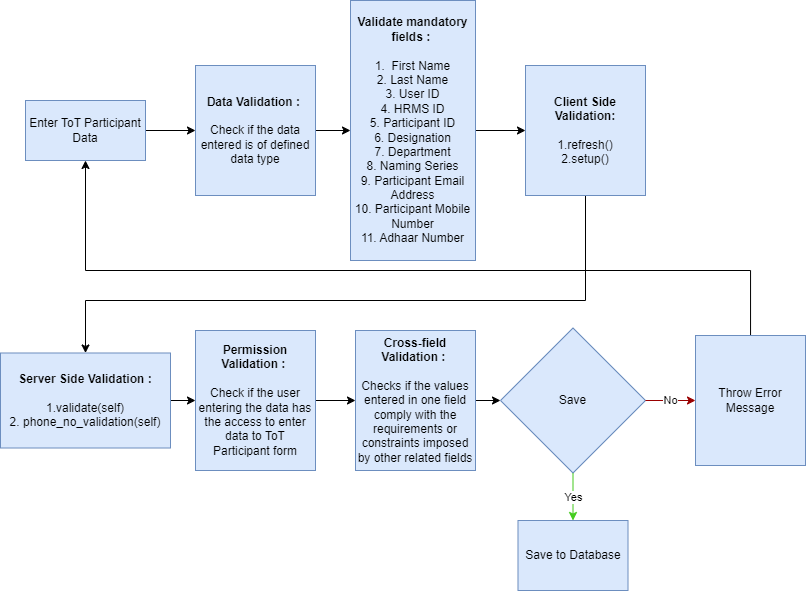
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Enabled | Check |  |  |
| 2 | First Name | Data | Yes |  |
| 3 | Middle Name | Data |  |  |
| 4 | Last Name | Data |  |  |
| 5 | User ID | Data |  |  |
| 6 | Joining Date | Date |  |  |
| 7 | HRMS ID | Data |  |  |
| 8 | Participant Id | Link |  | **Student** |
| 9 | Designation | Data |  |  |
| 10 | Department | Data |  |  |
| 11 | Name of the Institute | Data |  |  |
| 12 | Naming Series | Select |  | TOT-PART-.YYYY.- |
| 13 | Participant Email Address | Data | Yes | Email |
| 14 | Participant Mobile Number | Phone |  |  |
| 15 | Adhaar Number | Data |  |  |
| 16 | Participant Caste | Link |  | Student Category |
| 17 | Image | Attach Image |  |  |
| 18 | Date of Birth | Date |  |  |
| 19 | Blood Group | Select |  | A+  A-  B+  B-  O+  O-  AB+  AB- |
| 20 | Gender | Link | Yes | Gender |
| 21 | Nationality | Data |  |  |
| 22 | State | Link | Yes | State |
| 23 | District | Link | Yes | Districts |
| 24 | Block | Link | Yes | Blocks |
| 25 | Police Station | Data |  |  |
| 26 | Post Office | Data |  |  |
| 27 | Village | Data |  |  |
| 28 | Address Line 1 | Data |  |  |
| 29 | Address Line 2 | Data |  |  |
| 30 | Pincode | Data |  |  |
| 31 | Country | Link |  | Country |
| 32 | Participant Name | Data |  |  |
| 33 | Student No | Link |  |  |
| **Previous Participations** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 34 | Programs | Link |  | Programs |
| 35 | Academic Year | Link |  | Academic Year |
| 36 | Participant Selection | Link |  | ToT Participant Selection |
| 37 | Participant Enrollment | Link |  | ToT Participant Enrollment |
| 38 | Program Start Date | Data |  |  |
| 39 | Program End Date | Data |  |  |
| 40 | Semester | Link |  | Program |
| 41 | Academic Term | Link |  | Academic Term |
| **Participant Education Details** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 42 | Qualification | Data |  |  |
| 43 | Institute | Data |  |  |
| 44 | Board | Data |  |  |
| 45 | Total Marks | Data |  |  |
| 46 | Earned Marks | Data |  |  |
| 47 | Percentage | Data |  |  |
| 48 | Year of Completion | Data |  |  |
| **Participant Experience Details** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 49 | Company Name | Data |  |  |
| 50 | Job Profile | Data |  |  |
| 51 | Job Type | Data |  |  |
| 52 | Job Start Date | Date |  |  |
| 53 | Job End Date | Date |  |  |
| 54 | Job Duration | Date |  |  |
| **Participant Medical History** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 55 | Illness | Data |  |  |
| 56 | Description | Data |  |  |
| 57 | Attach Prescription | Data |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

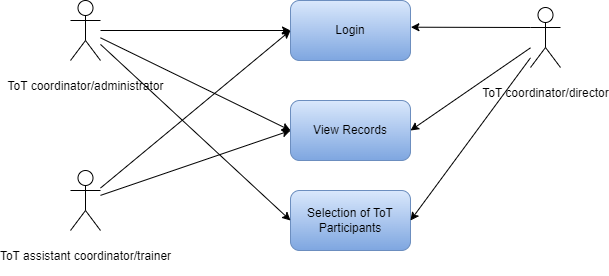
### Pre-requisites and Dependency

* Not Applicable

## ToT Participant Selection

The ToT Participant Selection screen is to be used by the ToT officials for Selecting Participants for a ToT program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Course Type, Programs, Semester, Academic Year, ToT Participant Batch, Posting Date, Academic Year, TOT Start Date, TOT End Date.
* Fields displayed on condition :
* Linked Fields : The Course Type field is linked to the Program Grades screen.The Program field is linked to the Programs screen.The Semester field is linked to the Program screen.The Academic Year field is linked to the Academic Year screen.The Academic Term field is linked to the Academic Term Screen.The Participant Field is linked to the ToT Participant Screen.The room type field is linked with the Room Type screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) - This function when triggered validates the chosen academic year in the screen by checking whether the academic year has started on not.
* On\_submit(self) - This function when triggered checks whether the child table is empty or not along with updating particpant details.
* Before\_submit(self) - This function triggers the create\_tot\_batch function when triggered.
* Create\_tot\_batch(self) - This function creates a ToT batch and saves the record.
* Check\_program\_enrollment(self) - This function when triggered checks whether a participant that has been selected has already been enrolled in the same course for the same ToT program or not.

**Notification:**

* Not Applicable

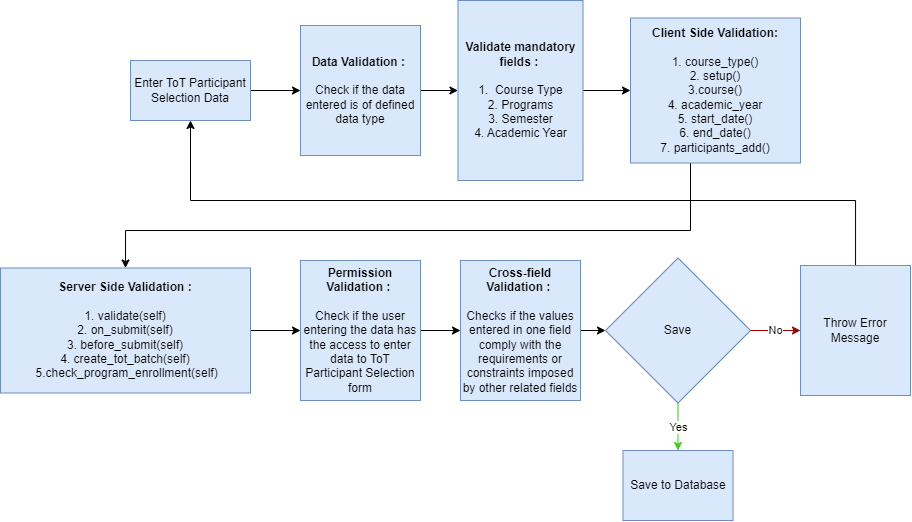
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Course Type | Link | Yes | Program Grades |
| 2 | Program | Link | Yes | Programs |
| 3 | Semester | Link |  | Program |
| 4 | ToT Participant Batch | Data | Yes |  |
| 5 | Posting Date | Date | Yes |  |
| 6 | Academic Year | Link | Yes | Academic Year |
| 7 | Academic Term | Link |  | Academic Term |
| 8 | TOT Start Date | Date | Yes |  |
| 9 | TOT End Date | Date | Yes |  |
| 10 | Naming Series | Select |  | TOT-PART-SEL-.YYYY.- |
| **Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 11 | Participant | Link |  |  |
| 12 | Participant Name | Data |  | ToT Participant |
| 13 | HRMS ID | Data |  |  |
| 14 | State | Data |  |  |
| 15 | District | Data |  |  |
| 16 | Pincode | Data |  |  |
| 17 | Gender | Data |  |  |
| 18 | Department | Data |  |  |
| 19 | Designation | Data |  |  |
| 20 | Mobile Number | Data |  |  |
| 21 | Email Address | Data |  |  |
| 22 | IS Hostel Required | Check |  |  |
| 23 | Room Type | Link |  | Room Type |
| 24 | Reporting Status | Data |  |  |
| 25 | Participant Enrollment No | Data |  | ToT Participant Enrollment |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

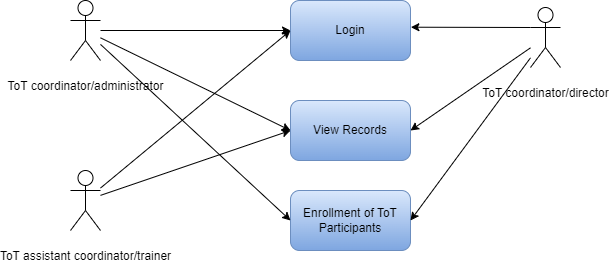
### Pre-requisites and Dependency

* Not Applicable

## ToT Participant Enrollment

The ToT Participant Enrollment screen is to be used by the ToT officials for enrolling the participants who have been selected for a particular ToT program into their respective courses.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : ToT Participant Selection ID
* Fields displayed on condition :
* Linked Fields : The ToT Participant Select ID field is linked to the ToT Participant Selection Screen.The Programs field is linked to the Programs screen.The Semester field is linked to the Program screen.The Academic Year field is linked to the Academic Year screen.The Academic Term Field is linked to the Academic Term screen.The Participant ID field is linked to the ToT Participant Screen.The Room Type field is linked to the Room Type screen.
* Autofetched fields : Participant List.

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) -
* On\_submit(self) - This function when triggered calls the create\_participant(self) function.
* Create\_participant(self) - This function creates the record of a participant upon the submission of the form.
* On\_cancel(self) - This function when triggered cancels the record and other records associated with it.
* Enroll\_participant(self) - This function is used for enrolling participants into their respective courses.
* Participant\_count\_validation(self) - This function when triggerd validates the number of participants that are being enrolled in the current record.
* Make\_participant(tot\_participant\_enrollment) - This function is used for making a participant if record for a participant being enrolled doesn’t exist.
* Make\_enrollment(tot\_participant\_enrollment) - This function when triggered enrolls the participants whose details have been entered in the current screen.
* Get\_participants(participant\_selection\_id) - This function is used to get the records of all the participants from Selected participants list based on the arguments that it receives.
* Tot\_participant\_selection\_id(doctype, txt, searchfield, start, page\_len, filters) - This function when triggered fetches participant selection id based on filter parameters that it receives as arguments.

**Notification:**

* Not Applicable

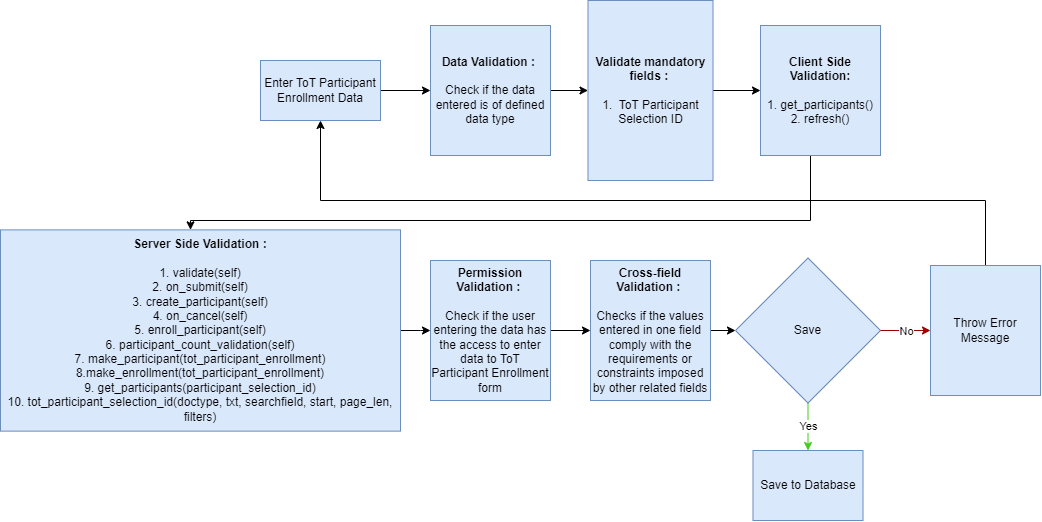
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | ToT Participant Selection ID | Link | Yes | ToT Participant Selection |
| 2 | Programs | Link |  | Programs |
| 3 | Semester | Link |  | Program |
| 4 | ToT Participant Batch | Data |  |  |
| 5 | Academic Year | Link |  | Academic Year |
| 6 | Academic Term | Link |  | Academic Term |
| 7 | TOT Start Date | Date |  |  |
| 8 | TOT End Date | Date |  |  |
| 9 | Naming Series | Select |  | TOT-PAR-ENR-.YYYY.- |
| 10 | Participant Creation Status | Select |  | In Process  Failed  Successful |
| 11 | Participant Enrollment Status | Select |  | In Process  Failed  Successful |
| 12 | Get Participants | Button |  |  |
| 13 | Status | Data |  |  |
| **Participant List** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 14 | First Name | Data |  |  |
| 15 | Middle Name | Data |  |  |
| 16 | Last Name | Data |  |  |
| 17 | Participant ID | Link |  | ToT Participant |
| 18 | Participant Name | Data |  |  |
| 19 | HRMS ID | Data |  |  |
| 20 | Odisha | Data |  |  |
| 21 | District | Data |  |  |
| 22 | Pincode | Data |  |  |
| 23 | Mobile Number | Phone |  |  |
| 24 | Email Address | Data |  |  |
| 25 | Reported | Check |  |  |
| 26 | Is Hostel Required | Check |  |  |
| 27 | Room Type | Link |  | Room Type |
| 28 | Program Enrollment | Data |  | Program Enrollment |

### Processes After Form Submission

* Not Applicable

### Process Flow:



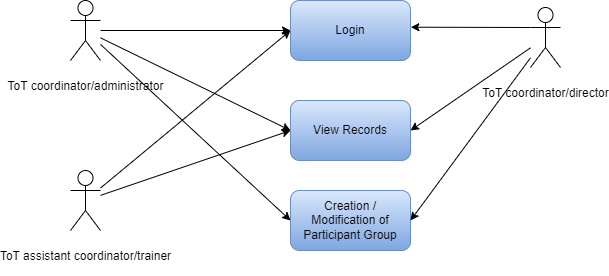
### Pre-requisites and Dependency

* Not Applicable

## Participant Group

The Participant Group screen is to be used by the ToT officials for creating participant groups for a particular module of a Course of a ToT program and scheduling classes for that module.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Academic Year, Participant Category, Academic Term, Participant Group, Course.
* Fields displayed on condition :
* Linked Fields : The Participant Enrollment ID field is linked with the ToT Participant Enrollment Screen.The Academic Year field is linked to the Academic Year screen.The Programs field is linked to the Programs screen.The Academic Term field is linked to the Academic Term Screen.The Module ID field is linked to the Course screen.The Instructors field is linked to the Instructor screen.The Room Name field is linked to the Room screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) - This function when triggered creates new records in the ToT Class Schedule screen.
* Calculate\_total\_hours(self) - This function when triggered calculates the duration between the start and end times.
* Participant(doctype, txt, searchfield, start, page\_len, filters) - This function when triggered fetches participant records based on the parameters that it receives.
* Get\_enrollment\_details(enrollment\_id) - This function when triggered fetches details from the ToT Participant Enrollment screen record corresponding with the enrollment id that the function receives as a parameter.
* get\_module\_name(module\_id) - This function when triggered gets the module name corresponding to the module id.
* Get\_participants(enrollment\_id) -This function when triggered fetches the details of all the students from the Program Enrollment table based on the arguments that it receives as parameters.

**Notification:**

* Not Applicable

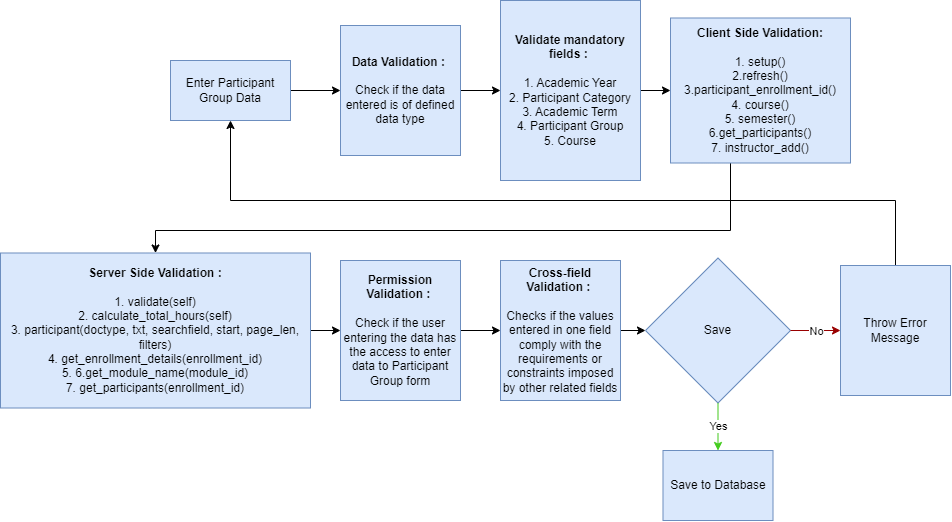
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Disabled | Check |  |  |
| 2 | Participant Enrollment ID | Link |  | ToT Participant Enrollment |
| 3 | Academic Year | Link |  | Academic Year |
| 4 | Programs | Link |  | Programs |
| 5 | Participant Group Name | Data |  |  |
| 6 | Group Based On | Select |  | Course |
| 7 | Academic Term | Link |  | Academic Term |
| 8 | Semester | Data |  |  |
| 9 | Module ID | Link |  | Course |
| 10 | Batch | Data |  |  |
| 11 | Module Name | Data |  |  |
| 12 | Get Participants | Button |  |  |
| **Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 13 | Participant | Data |  |  |
| 14 | Participant Name | Data |  |  |
| 15 | Active | Check |  |  |
| **Instructor** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 16 | Instructors | Link |  | Instructor |
| 17 | Instructor Name | Data |  |  |
| **Classes** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 18 | Scheduled Date | Date |  |  |
| 19 | Scheduled Time | Time |  |  |
| 20 | Room Name | Link |  | Room |
| 21 | Room Number | Data |  |  |
| 22 | From Time | Time |  |  |
| 23 | To Time | Time |  |  |
| 24 | Duration | Data |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

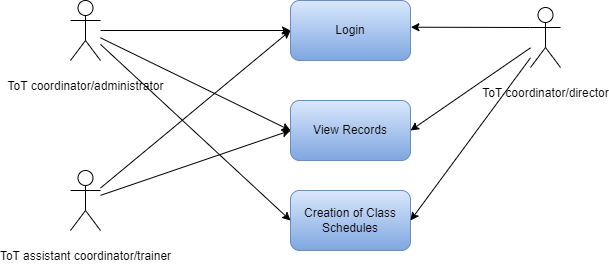
### Pre-requisites and Dependency

* Not Applicable

## ToT Class Schedule

The ToT Class Schedule screen is used for holding the class schedules for all the modules of a ToT course.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Not Applicable
* Fields displayed on condition :
* Linked Fields : Not Applicable
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Not Applicable

**Notification:**

* Not Applicable

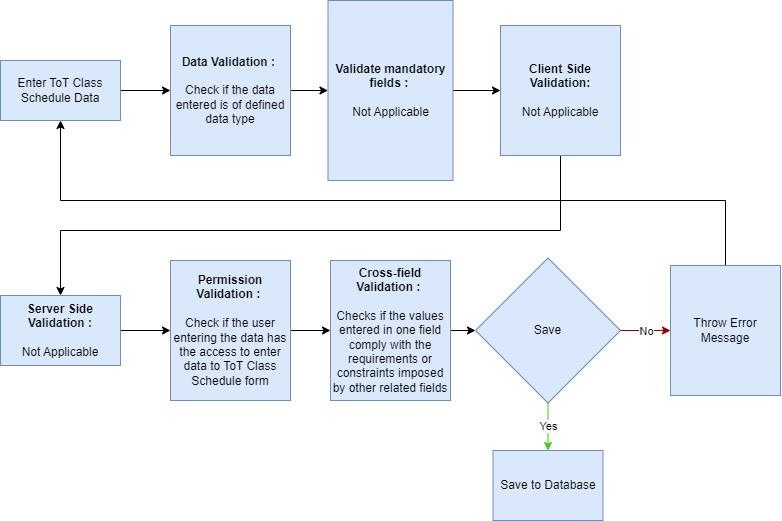
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group ID | Data |  |  |
| 2 | Academic Term | Data |  |  |
| 3 | Academic Year | Data |  |  |
| 4 | Course ID | Data |  |  |
| 5 | Module ID | Data |  |  |
| 6 | Scheduled Date | Data |  |  |
| 7 | Room Number | Data |  |  |
| 8 | From Time | Data |  |  |
| 9 | Duration | Data |  |  |
| 10 | Course Name | Data |  |  |
| 11 | Module Name | Data |  |  |
| 12 | Scheduled Time | Data |  |  |
| 13 | Room Name | Data |  |  |
| 14 | To Time | Data |  |  |
| **Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 15 | Participant ID | Data |  |  |
| 16 | Participant Name | Data |  |  |
| 17 | Active | Check |  |  |
| **Trainers** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 18 | Trainer ID | Data |  |  |
| 19 | Trainer Name | Data |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

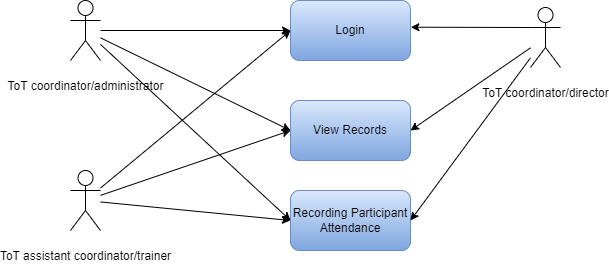
### Pre-requisites and Dependency

* Not Applicable

## ToT Participant Attendance

The ToT Participant Attendance screen is to be used by the ToT officials to mark the attendance of participants individually during a class of a module in a ToT Program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Instructor ID, Participant ID, Status
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen.The Instructor ID Field is linked with the Instructor screen.The Participant ID field is linked with the ToT Participant Screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Instructor(doctype, txt, searchfield, start, page\_len, filters) - This function when triggered fetches instructor records based on the parameters that it receives.
* participant(doctype, txt, searchfield, start, page\_len, filters) -This function when triggered fetches participant records based on the parameters that it receives.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) - This function when triggered fetches the name of an instructor from the Instructor Table table based on the participant group id and the instructor id that it receives as parameters.
* Get\_participant\_name(participant\_group\_id, participant\_name) - This function when triggered fetches the name of an participant from the Participant Table table based on the participant group id and the participant id that it receives as parameter.

**Notification:**

* Not Applicable

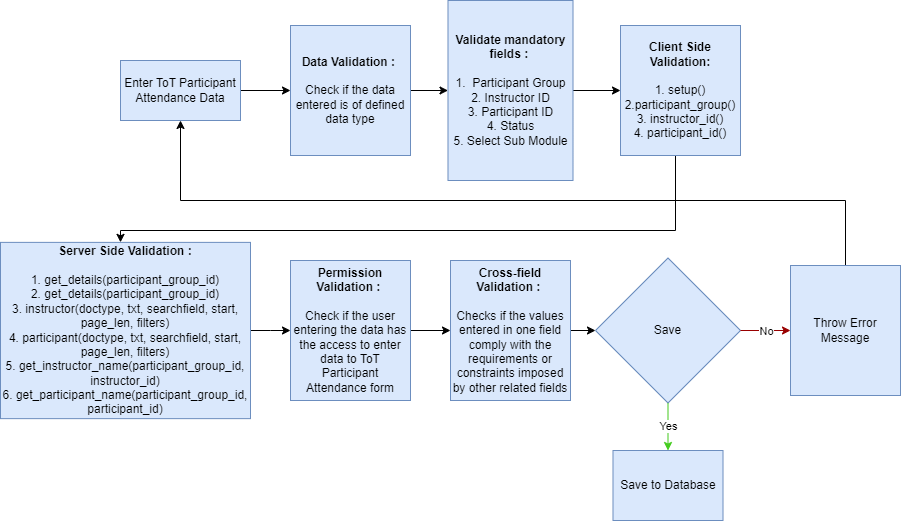
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group | Link |  | Participant Group |
| 2 | Select Module | Data |  |  |
| 3 | Select Course | Data |  |  |
| 4 | Academic Year | Data |  |  |
| 5 | Instructor ID | Link |  | Instructor |
| 6 | Participant ID | Link |  | ToT Participant |
| 7 | Date | Select |  |  |
| 8 | Status | Select |  | Present  Absent |
| 9 | Academic Year | Data |  |  |
| 10 | Instructor Name | Data |  |  |
| 11 | Participant Name | Data |  |  |
| 12 | Time | Time |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

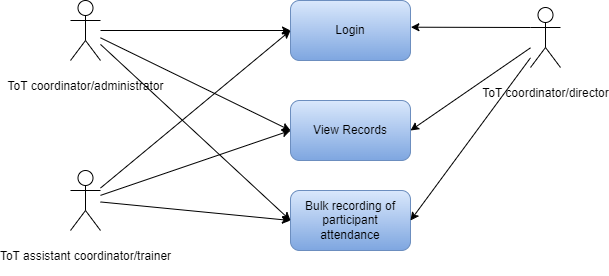
### Pre-requisites and Dependency

* Not Applicable

## Participant Attendance Tool

The ToT Participant Attendance Tool screen is to be used by the ToT officials to mark the attendance of all the participants of a participant group in bulk during a class of a module in a ToT Program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Based On, Participant Group, Instructor ID, Date, Get Participants
* Fields displayed on condition :
* Linked Fields : Not Applicable
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) - This function when triggered creates new records in the ToT Participant Attendance Screen.
* Get\_participant\_group(based\_on) - This function when triggered fetches the details of participant group names based on the parameter that it received.
* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) -This function when triggered gets the name of an instructor based on the participant group id and the instructor id that it receives as parameters.
* Get\_participants(participant\_group\_id) - This function when triggered gets the details of the participant belonging to a participant group based on the participant group id that it receives as a parameter.

**Notification:**

* Not Applicable

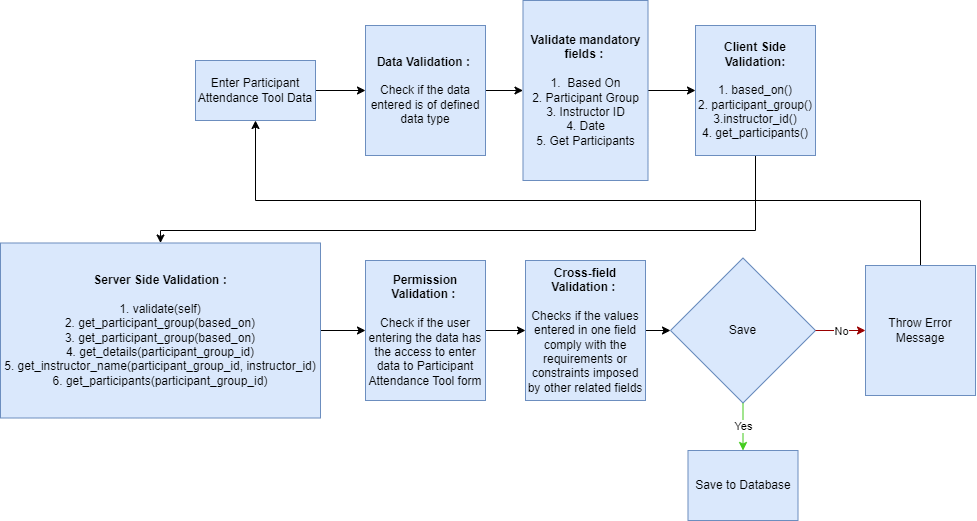
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Based On | Select |  | Participant Group |
| 2 | Academic Year | Data |  |  |
| 3 | Select Course | Data |  |  |
| 4 | Participant Group | Select |  |  |
| 5 | Academic Term | Data |  |  |
| 6 | Select Module | Data |  |  |
| 7 | Instructor ID | Select |  |  |
| 8 | Date | Select |  |  |
| 9 | Instructor Name | Data |  |  |
| 10 | Time | Time |  |  |
| 11 | Get Participants | Button |  |  |
| **Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 12 | Participant ID | Data |  |  |
| 13 | Participant Name | Data |  |  |
| 14 | Present | Check |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:



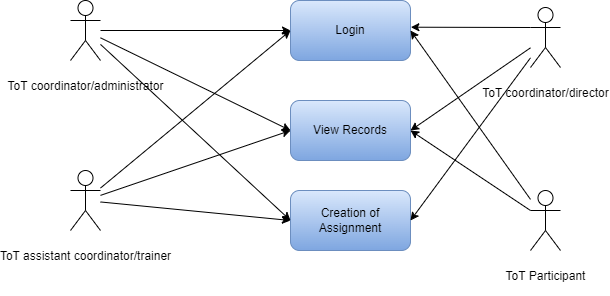
### Pre-requisites and Dependency

* Not Applicable

## Assignment

The Assignment screen is to be used by the ToT officials to allocate assignments to participant groups of a ToT program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Instructor ID, Assignment Name, Assessment Criteria, Assignment Attach Button
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) - This function when triggered fetches the name of an instructor based on the participant group id and the instructor id that it receives as a parameter.
* Get\_criteria\_details(course, assessment\_criteria) -This function when triggered fetches the assessment criteria details based on the course id and the assessment criteria id that it receives as a parameter.

**Notification:**

* Not Applicable

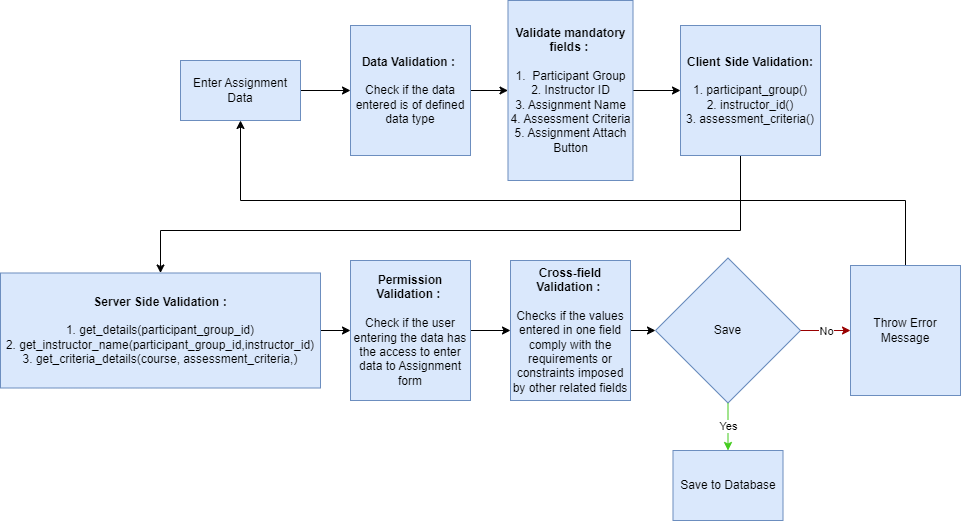
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Evaluate | Check |  |  |
| 2 | Participant Group | Link |  | Participant Group |
| 3 | Select Course | Data |  |  |
| 4 | Assignment Name | Data |  |  |
| 5 | Select Module | Data |  |  |
| 6 | Academic Year | Data |  |  |
| 7 | Instructor ID | Select |  |  |
| 8 | Assessment Criteria | Select |  |  |
| 9 | Weightage | Percent |  |  |
| 10 | Academic Term | Data |  |  |
| 11 | Instructor Name | Data |  |  |
| 12 | Total Marks | Data |  |  |
| 13 | Passing Marks | Data |  |  |
| 14 | Assignment Attach Button | Attach |  |  |
| 15 | Description | Long Text |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

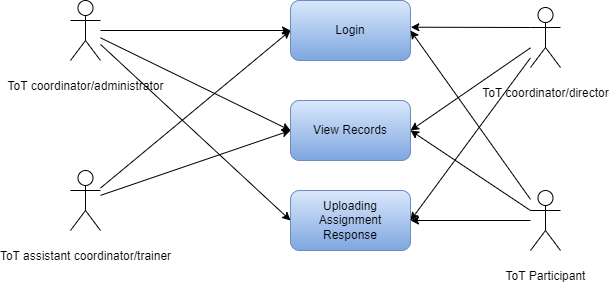
### Pre-requisites and Dependency

* Not Applicable

## Assignment Upload

The Assignment Upload screen is to be used by the ToT participants to upload their responses individually to the assignments allocated to them.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields :Participant Group, Participant ID, Instructor ID, Assessment Components, Attach Assignment.
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) - This function when triggered fetches the name of an instructor based on the participant group id and the instructor id that it receives as a parameter.
* Get\_assignment\_details(assignment\_name) -This function when triggered fetches the details of an assignment based on the assignment\_name that it receives as a parameter.
* Get\_participant\_name(participant\_group\_id, participant\_id) - This function when triggered fetches the name of a participant based on the participant group id and the participant id that it receives as a parameter.

**Notification:**

* Not Applicable

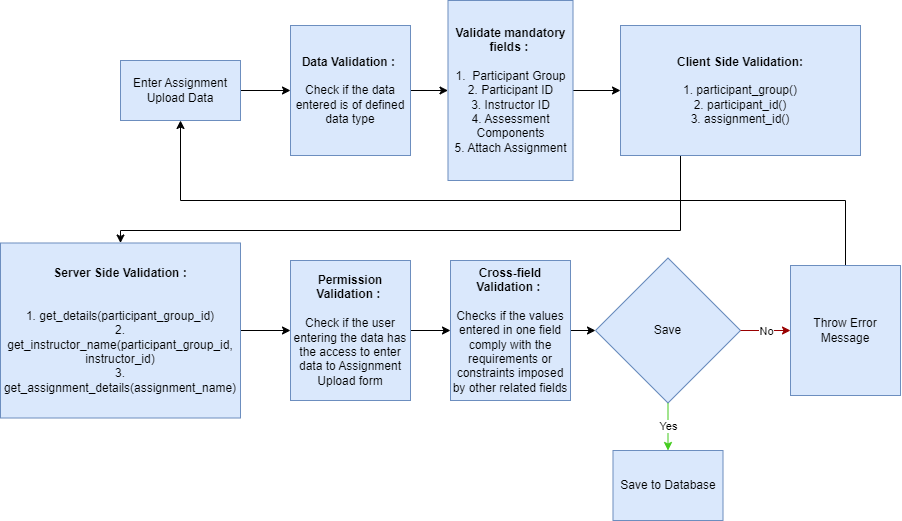
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group | Link |  | Participant Group |
| 2 | Select Module | Data |  |  |
| 3 | Select Course | Data |  |  |
| 4 | Academic Year | Data |  |  |
| 5 | Participant ID | Select |  |  |
| 6 | Assignment ID | Select |  |  |
| 7 | Assessment Component | Data |  |  |
| 8 | Weightage | Data |  |  |
| 9 | Academic Term | Data |  |  |
| 10 | Participant Name | Data |  |  |
| 11 | Assignment Name | Data |  |  |
| 12 | Total Marks | Data |  |  |
| 13 | Passing Marks | Data |  |  |
| 14 | Attach Assignment | Attach |  |  |
| 15 | Description | Long Text |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

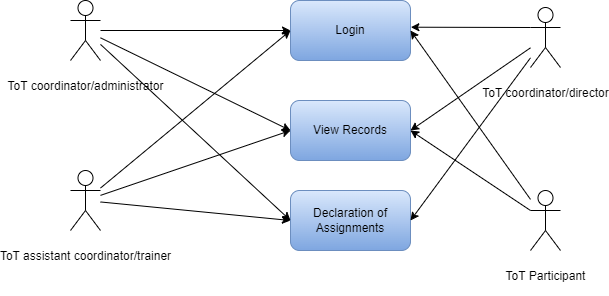
### Pre-requisites and Dependency

* Not Applicable

## Assignment Declaration

The Assignment Declaration screen is to be used by the ToT officials to create and declare assignments for a participant group of a ToT program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Assignment Name, Assignment Start Date, Participant Group, Select Module, Select Assessment Criteria, Trainer ID, Get Participants.
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen.The Assessment Criteria Field is linked with the Assessment Criteria.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) - This function when triggered fetches the name of an instructor based on the participant group id and the instructor id that it receives as a parameter.
* Get\_criteria\_details(course, assessment\_criteria) - This function when triggered gets the details of assessment criteria based on the parameters that it receives.
* Get\_participants(participant\_group\_id, attendance\_applicable, attendance\_percentage) - This function when triggered fetches the details of participants belonging to a participant group based on the participant group id that it receives as a parameter.

**Notification:**

* Not Applicable

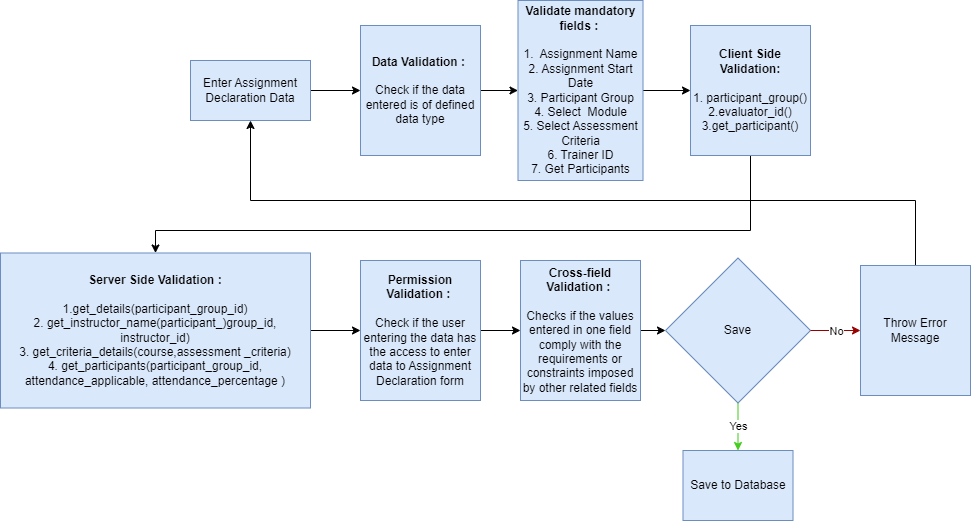
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Attendance Applicable | Check |  |  |
| 2 | Participant Group | Link |  | Participant Group |
| 3 | Assignment Start Date | Date |  |  |
| 4 | Academic Year | Data |  |  |
| 5 | Select Course | Data |  |  |
| 6 | Select Assessment Criteria | Select |  |  |
| 7 | Weightage | Data |  |  |
| 8 | Assignment Name | Data |  |  |
| 9 | Assignment End Data | Date |  |  |
| 10 | Academic Term | Data |  |  |
| 11 | Select Module | Data |  |  |
| 12 | Total Marks | Data |  |  |
| 13 | Pass Marks | Data |  |  |
| 14 | Evaluator ID | Select |  |  |
| 15 | Attendance Percentage | Percent |  |  |
| 16 | Evaluator Name | Data |  |  |
| 17 | Get Participants | Button |  |  |
| **Participant List** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 18 | Participant ID | Data |  |  |
| 19 | Participant Name | Data |  |  |
| 20 | Participant Attendance | Data |  |  |
| 21 | Status | Data |  |  |
| **Job Sheet** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
|  | Job Sheet Name | Data |  |  |
|  | Date | Date |  |  |
|  | From Time | Time |  |  |
|  | To Time | Time |  |  |
|  | Toral Duration(in hours) | Time |  |  |
|  | Assessment Criteria | Link |  | Assessment Critera |
|  | Total Marks | Data |  |  |
|  | Pass Marks | Data |  |  |
|  | Weightage | Data |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

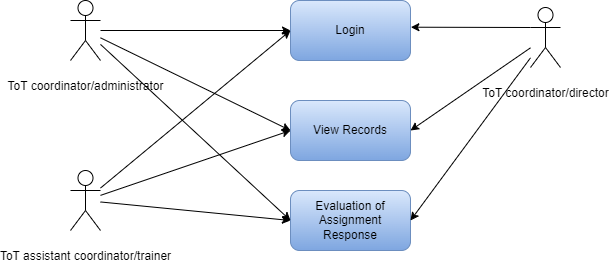
### Pre-requisites and Dependency

* Not Applicable

## Assignment Evaluation

The Assignment Evaluation screen is to be used by the ToT officials to evaluate performance of an individual participant for an assignment .

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Select Module, Instructor ID.
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen.The Assessment Component Field is linked with the Assessment Criteria.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) - This function when triggered fetches the name of an instructor based on the participant group id and the instructor id that it receives as a parameter.
* Get\_participant\_name(participant\_group\_id, participant\_id) -This function when triggered fetches the name of a participant based on the participant group id and the participant id that it receives as a parameter.
* Get\_assignment\_list(instructor\_id, participant\_group\_id, programs, course) - This function when triggered fetches the names of all the assignments belonging to that participant group, based on the received parameters.
* Get\_assignment\_details(assignment\_name) - This function when triggered fetches the details of an assignment based on the assignment\_name that it receives as a parameter.
* Set\_marks(participant\_id, assignment\_name) - This function when triggered checks whether a participant has uploaded a response for an assignment or not and sets a default marks for any participant who has not uploaded a response.

**Notification:**

* Not Applicable

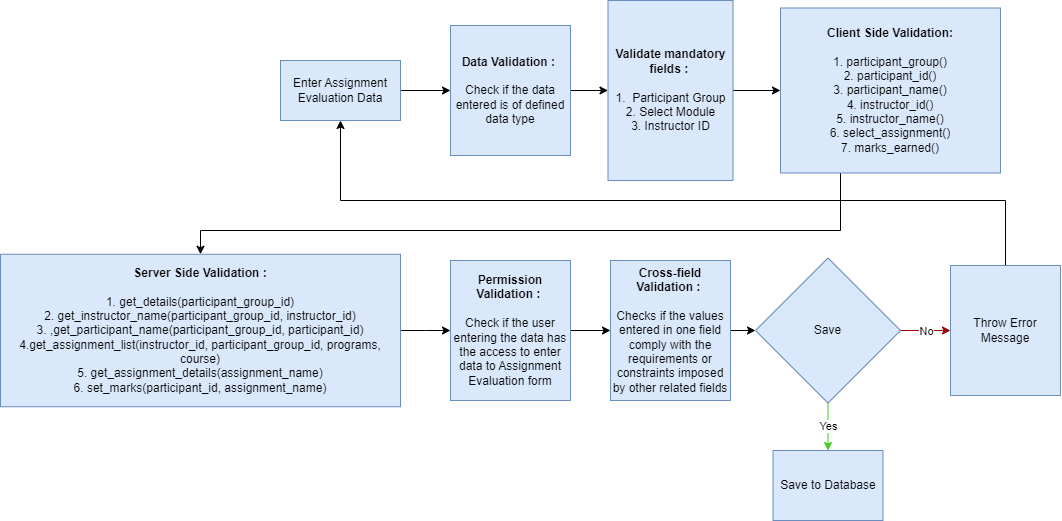
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group | Link |  | Participant Group |
| 2 | Select Module | Data |  |  |
| 3 | Select Course | Data |  |  |
| 4 | Academic Year | Data |  |  |
| 5 | Evaluator ID | Select |  |  |
| 6 | Select Assignment | Select |  |  |
| 7 | Participant ID | Select |  |  |
| 8 | Assessment Component | Link |  | Assessment Criteria |
| 9 | Academic Term | Data |  |  |
| 10 | Evaluator Name | Data |  |  |
| 11 | Assignment Name | Data |  |  |
| 12 | Participant Name | Data |  |  |
| 13 | Total Marks | Int |  |  |
| 14 | Weightage | Data |  |  |
| 15 | Passing Marks | Data |  |  |
| 16 | Marks Earned | Int |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:



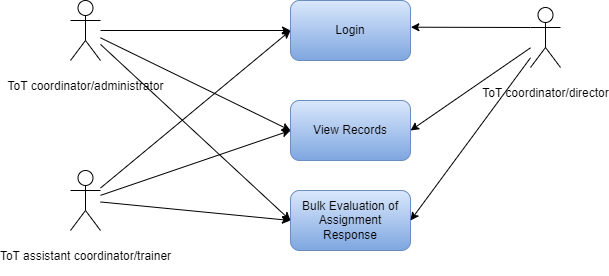
### Pre-requisites and Dependency

* Not Applicable

## Assignment Evaluation Tool

The assignment Evaluation Tool screen is to be used by the ToT officials to evaluate the performances of all participants of a participant group for a particular assignment allocated to that participant group.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Instructor ID, Job Sheet/Assessment, Get Participants.
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen.The Job Sheet/Assessment Scheme field is linked with the Job Sheet screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) - This function when triggered creates records of the evaluated assignments in the assignment evaluation screen.
* Get\_details(participant\_group\_id) -This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_instructor\_name(participant\_group\_id, instructor\_id) - This function when triggered fetches the name of an instructor based on the participant group id and the instructor id that it receives as parameters.
* Get\_Assignment\_list(instructor\_id, participant\_group\_id, programs, course) - This function when triggered fetches the assignments belonging to a participant group based on the parameters that it receives.
* Get\_assignment\_details(assignment\_name) - This function when triggered fetches the details of an assignment based on the assignment id that it receives as a parameter.

**Notification:**

* Not Applicable

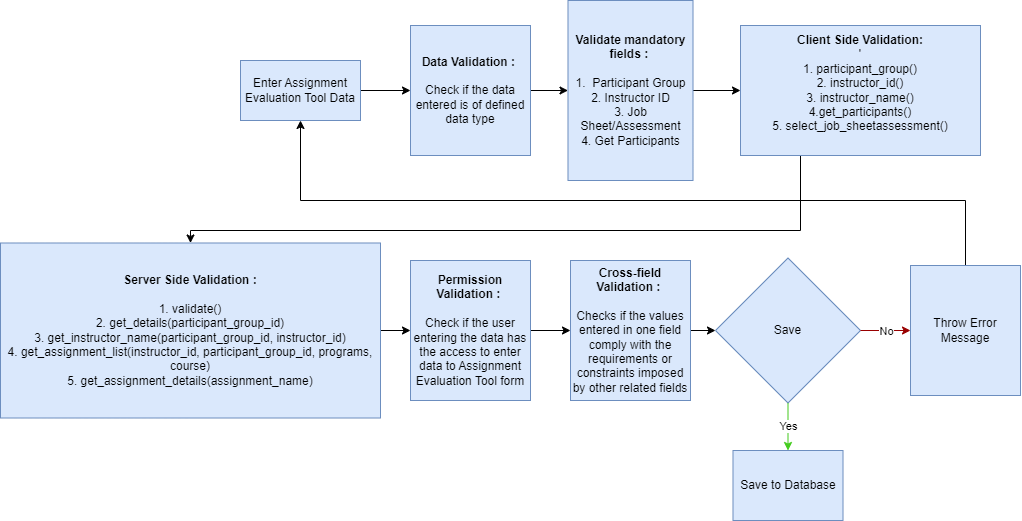
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group | Link |  | Participant Group |
| 2 | Academic Year | Data |  |  |
| 3 | Evaluator ID | Select |  |  |
| 4 | Select Course | Data |  |  |
| 5 | Total Participants | Data |  |  |
| 6 | Acdemic Term | Data |  |  |
| 7 | Evaluator Name | Data |  |  |
| 8 | Select Module | Data |  |  |
| 9 | Select Job Sheet/Assessment | Select |  |  |
| 10 | Total Marks | Data |  |  |
| 11 | Weightage | Data |  |  |
| 12 | Job Sheet/Assessment Scheme | Link |  | Job Sheet |
| 13 | Assignment Name | Data |  |  |
| 14 | Passing Marks | Data |  |  |
| 15 | Assessment Criteria | Data |  |  |
| 16 | Get Participants | Button |  |  |
| **Qualified Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 17 | Participant ID |  |  |  |
| 18 | Participant Name |  |  |  |
| 19 | Earned Marks |  |  |  |
| **Disqualified Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 20 | Participant ID |  |  |  |
| 21 | Participant Name |  |  |  |
| 22 | Earned Marks |  |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

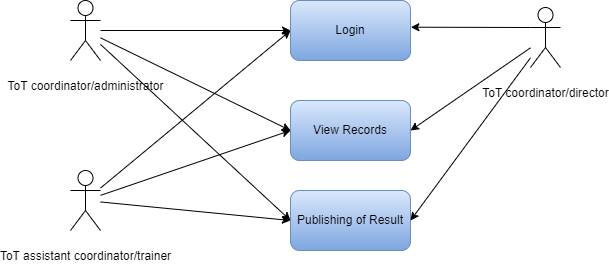
### Pre-requisites and Dependency

* Not Applicable

## Final Assignment Result

The Final Assignment Result Screen is to be used by the ToT officials for declaring the results of participants of a Participant Group.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Participant ID, Grading Scale, Assessment Status
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen. The grading scale field is linked to the Grading scale screen. The Assessment Criteria Field is linked to the Assessment Criteria screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Get\_details(participant\_group\_id) -This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_participant\_details(participant\_group\_id, participant\_id) - This function when triggered fetches the name of participant based on the participant id and the participant group id values that it receives as parameters and calculates the participants attendance before returning the name and attendance person of the participant.
* Get\_assignments(participant\_group\_id, participant\_id, grading\_scale) -This function when triggered gets the data of all evaluated assignments of a participant belonging to a participant group.

**Notification:**

* Not Applicable

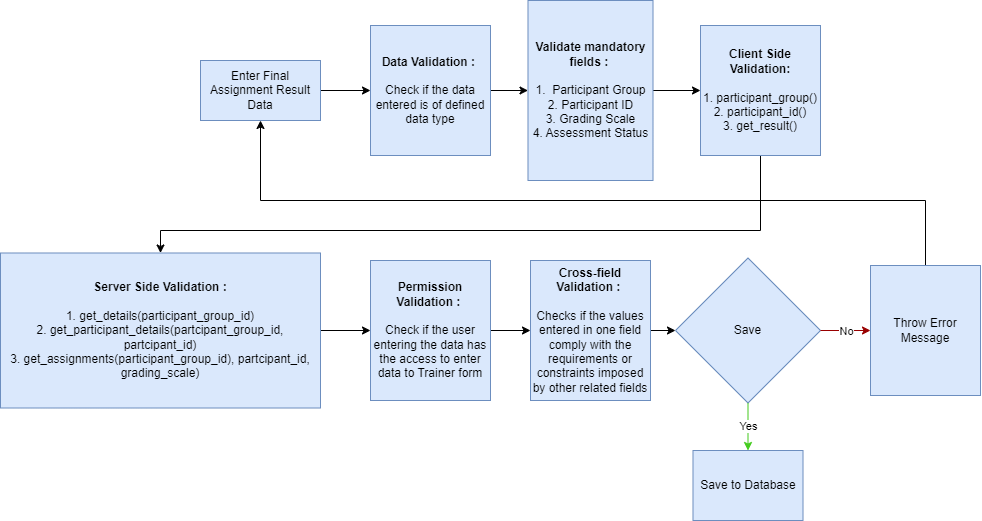
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group | Link |  | Participant Group |
| 2 | Course Code | Data |  |  |
| 3 | Select Module | Data |  |  |
| 4 | Select Course | Data |  |  |
| 5 | Course Name | Data |  |  |
| 6 | Academic Year | Data |  |  |
| 7 | Participant ID | Select |  |  |
| 8 | Grading Scale | Link |  | Grading Scale |
| 9 | Attendance Percentage | Data |  |  |
| 10 | Academic Term | Data |  |  |
| 11 | Participant Name | Data |  |  |
| 12 | Assessment Status | Data |  |  |
| 13 | Get Result | Button |  |  |
| 14 | Overall Percentage | Data |  |  |
| 15 | Overall Result | Data |  |  |
| 16 | Overall Grade | Data |  |  |
| 17 | Comment | Small Text |  |  |
| **Assessment Result Item** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 18 | Assignment ID | Data |  |  |
| 19 | Assignment Name | Data |  |  |
| 20 | Assessment Criteria | Link |  | Assessment Criteria |
| 21 | Earned Marks | Float |  |  |
| 22 | Total Marks | Float |  |  |
| 23 | Percentage | Percent |  |  |
| 24 | Grade | Data |  |  |
| 25 | Result | Select |  | PASS  FAIL |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

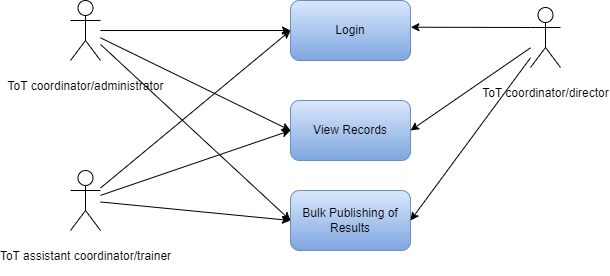
### Pre-requisites and Dependency

* Not Applicable

## Final Result Declaration Tool

The Final Result Declaration Tool is to be used by the ToT officials for generating results of all the participants of a participant group in a ToT program.

### Use Case Diagram



### Design Workflow

* Design worflow is not applicable to this screen as records in it are created by only one role and are not sent to users with other roles for approval or modification.

### Validations

**Client Side Validation :**

Client-side validation refers to the process of validating user input directly on the client's web browser before saving it to the server for further processing.

The following validations are done on client side for ToT Participant Screen:

* Mandatory checks for fields : Participant Group, Grading Scale, Get Participants
* Fields displayed on condition :
* Linked Fields : The Participant Group Field is linked with the Participant Group screen. The grading scale field is linked to the Grading scale screen.
* Autofetched fields : Not Applicable

**Server Side Validation :**

Server-side validation refers to the process of validating user input on the server to ensure data integrity, security, and adherence to business rules before accepting or processing the information.

The following validations are done on server side for ToT Participant Screen:

* Validate(self) - This function when triggered creates records in the final assignment screen.
* Get\_details(participant\_group\_id) - This function when triggered fetches the details of a participant group based on the participant group id value that it receives as a value.
* Get\_participants(participant\_group\_id) - This function when triggered fetches the details of participants belonging to a participant group based on the participant group id that it receives as a parameter.

**Notification:**

* Not Applicable

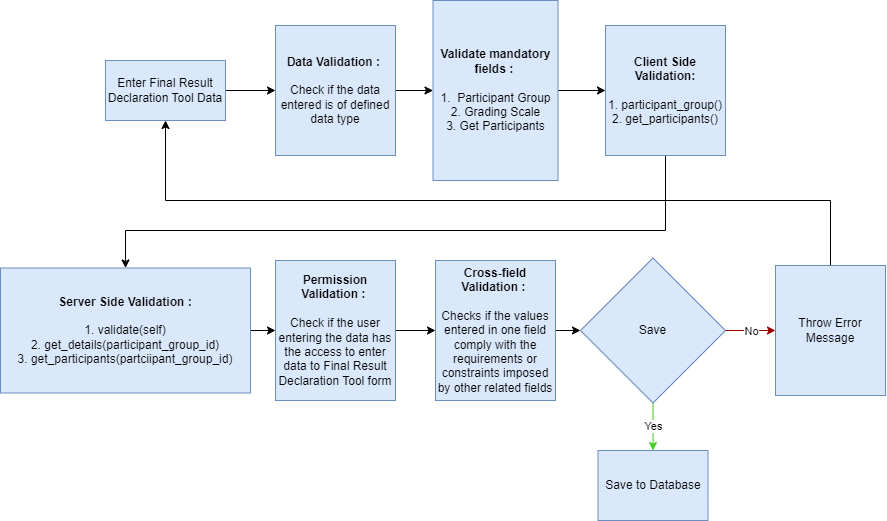
### Field List:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 1 | Participant Group | Link |  | Participant Group |
| 2 | Course COde | Data |  |  |
| 3 | Select Module | Data |  |  |
| 4 | Academic Year | Data |  |  |
| 5 | Grading Scale | Link |  | Grading Scale |
| 6 | Select Course | Data |  |  |
| 7 | Course Name | Data |  |  |
| 8 | No of Participants | Data |  |  |
| 9 | Academic Term | Data |  |  |
| 10 | Get Participants | Button |  |  |
| **Participants** | | | | |
| **ID** | **Field Name** | **Field Type** | **Mandatory** | **Options** |
| 11 | Participant ID | Data |  |  |
| 12 | Participant Name | Data |  |  |
| 13 | Participant Attendance | Percent |  |  |

### Processes After Form Submission

* Not Applicable

### Process Flow:

****

### Pre-requisites and Dependency

* Not Applicable

# Tools And Technologies

This section explains the tools that are used to build this design document

* Draw.io - it is an online tool used to develop the flowcharts used in this document.
* WPS office - it is a text editor that has been used to write and assemble this document

# Non-Functional Requirements

## Performance

Specify performance requirements, including response times, transaction throughput, and system scalability

## Reliability

Define the reliability requirements, such as availability, fault tolerance, and disaster recovery

## Usability

Describe usability requirements, including user-friendly interfaces, clear error messages, and intuitive workflows

## Compatibility

Specify compatibility requirements with different operating systems, web browsers, and devices

## Security

Security requirements, including authentication, access control, data encryption, and secure communication protocols

* Authentication: The system will provide robust and secure authentication mechanisms to ensure that only authorized users can access the system and its resources
* Authorization: The system will enforce access controls to restrict users' actions based on their roles and privileges, ensuring that users can only access the resources they are authorized to use
* Data Encryption: Sensitive data, both at rest and in transit, will be encrypted to protect it from unauthorized access or interception

Secure Communication: Secure communication protocols (e.g., HTTPS, TLS) should be used to protect data exchanged between clients and servers

## Compliance

Specify any legal or regulatory compliance requirements, such as GDPR, PCI DSS, or other industry-specific standards

## Documentation

Describe the documentation requirements, including user manuals, developer guides, and API documentation

## Security: User access and authorization

The screens and the data in them is made secure by creating roles with different permissions and then assigning those roles the appropriate users. For this screen following are the roles and their corresponding permissions:

### 

**Trainer**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |

**ToT Participant**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |
| 4 | ToT participant | Yes | No | No | No | NA | NA | NA |

**ToT Participant Selection**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |

**ToT Participant Enrollment**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |

**Participant Group**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |

**ToT Class Schedule**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |

**ToT Participant Attendance**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |

**Participant Attendance Tool**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |

**Assignment**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |
| 4 | ToT participant | Yes | No | No | No | NA | NA | NA |

**Assignment Upload**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | No | No | No | NA | NA | NA |
| 4 | ToT participant | Yes | Yes | Yes | No | NA | NA | NA |

**Assignment declaration**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |
| 4 | ToT participant | Yes | No | No | No | NA | NA | NA |

**Assignment Evaluation**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |

**Assignment Evaluation Tool**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |

**Final Assignment Result**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |

**Final Result Declaration Tool**

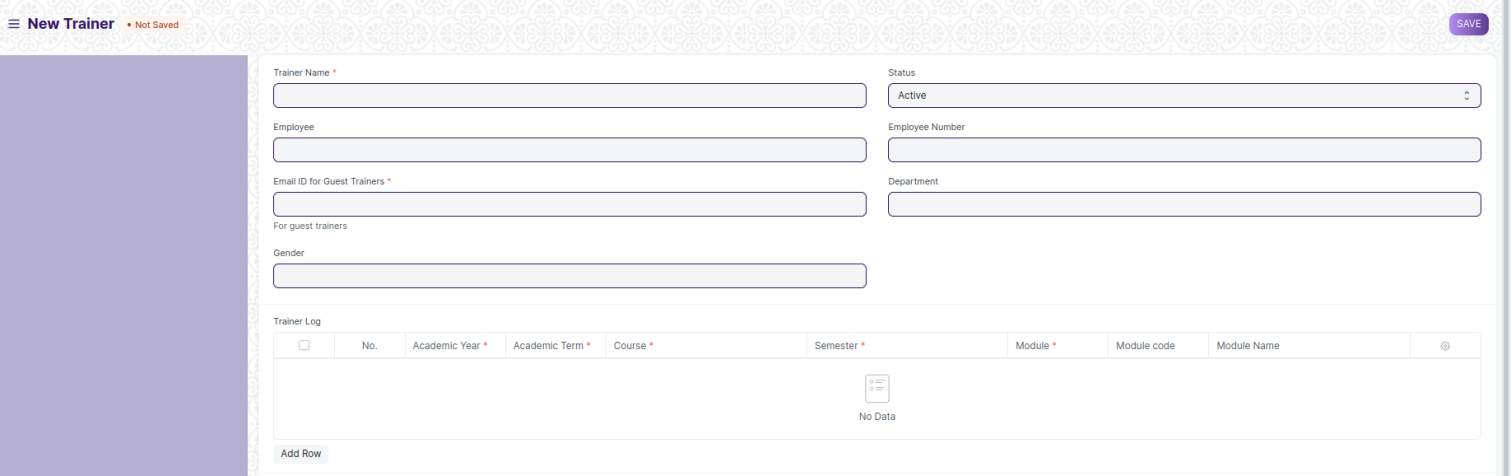
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Role** | **Read** | **Write** | **Create** | **Delete** | **Submit** | **Cancel** | **Amend** |
| 1 | ToT coordinator/director | Yes | Yes | Yes | No | NA | NA | NA |
| 2 | ToT coordinator/administrator | Yes | Yes | Yes | No | NA | NA | NA |
| 3 | ToT assistant coordinator/trainer | Yes | Yes | Yes | No | NA | NA | NA |

**Prototype**

# Prototype

The screen samples that have been derived based on the requirements gathered from the users / SME’s of the **Training Of Trainers** module

## Trainer



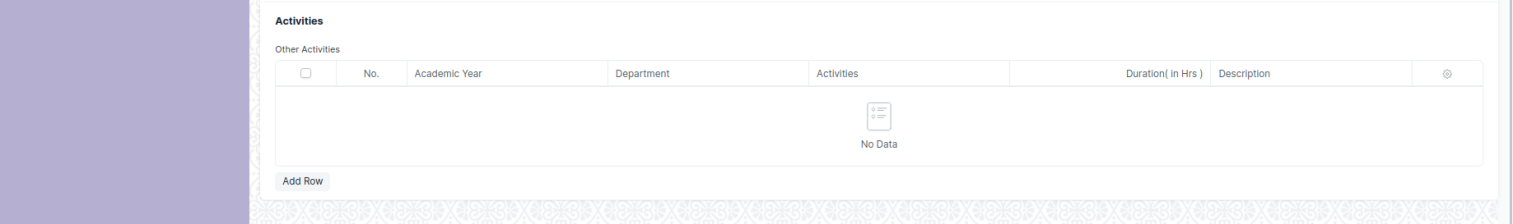
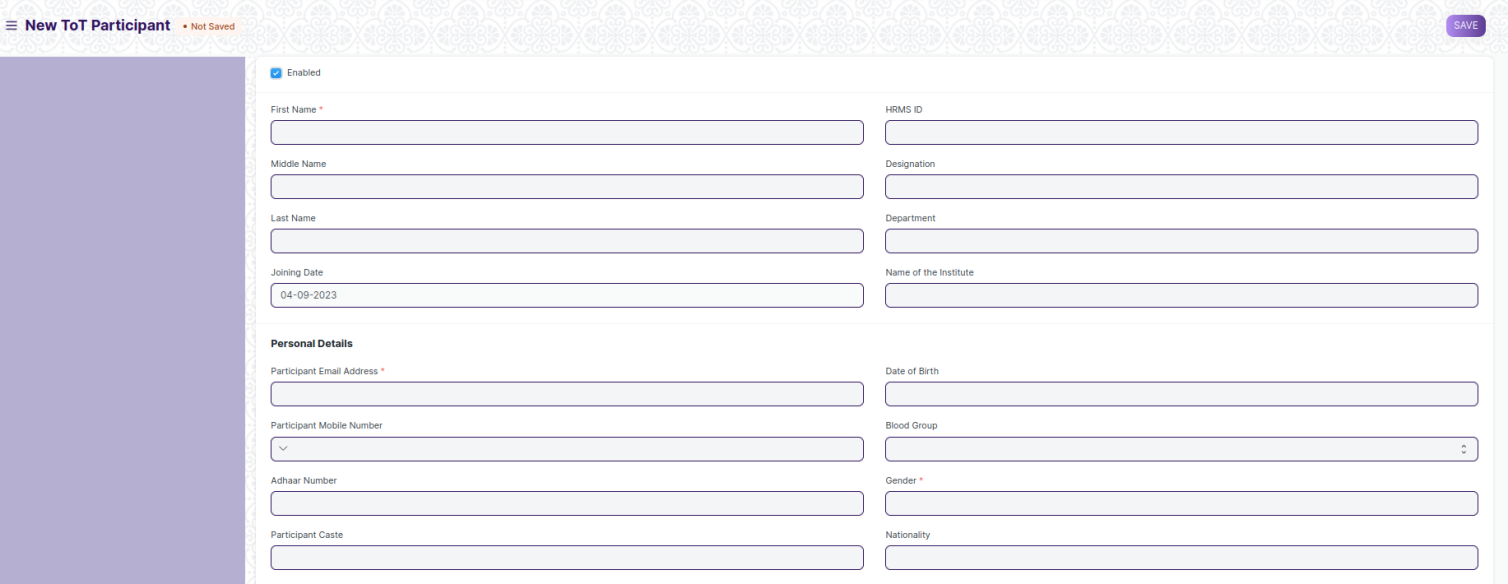
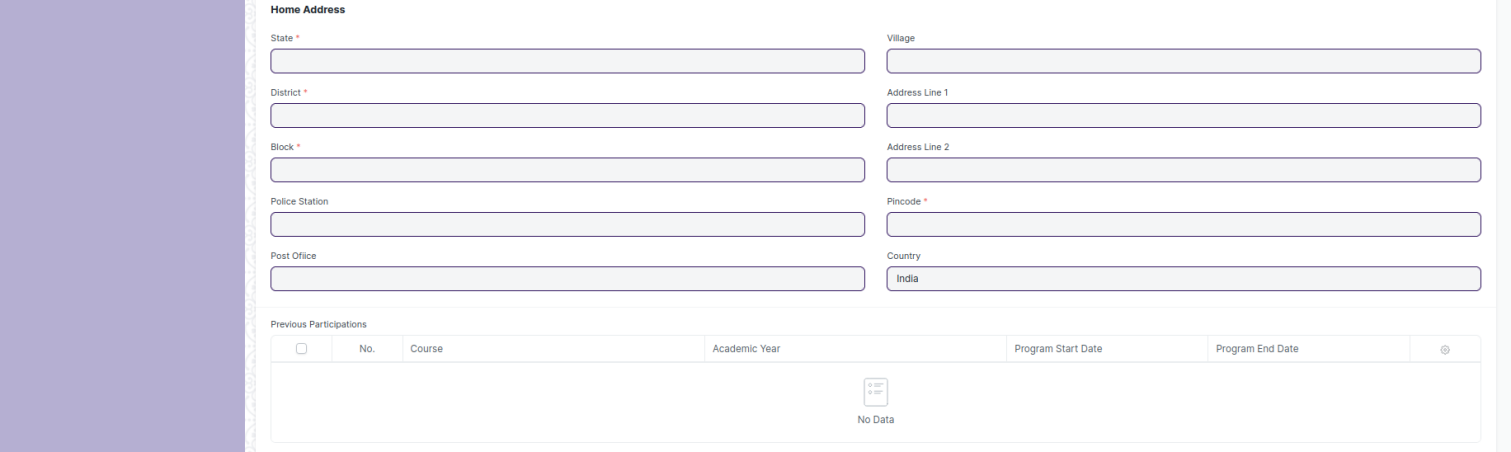


Figure 1 Trainer

## ToT Participant

Description: A ToT Participant is an individual who is qualified to take part in ToT Events.





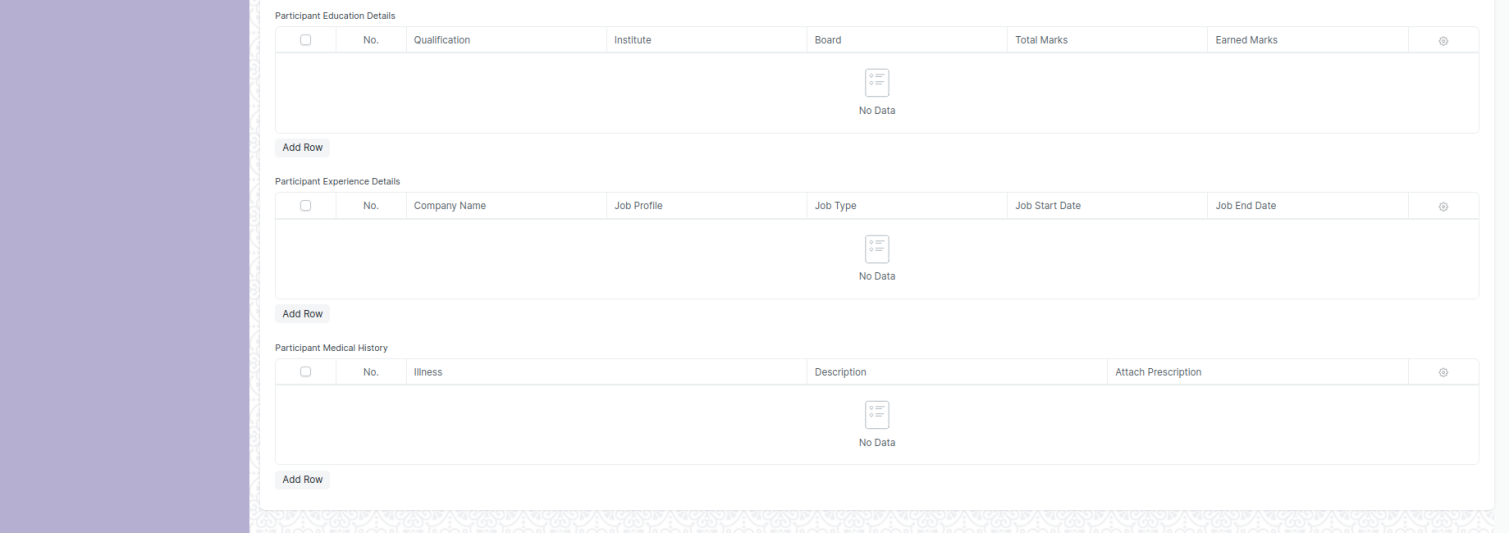


Figure 2 ToT Participant

## ToT Participant Selection

Description: This screen is provided to the user for selecting participants for a ToT Event from the pool of recorded eligible participants.

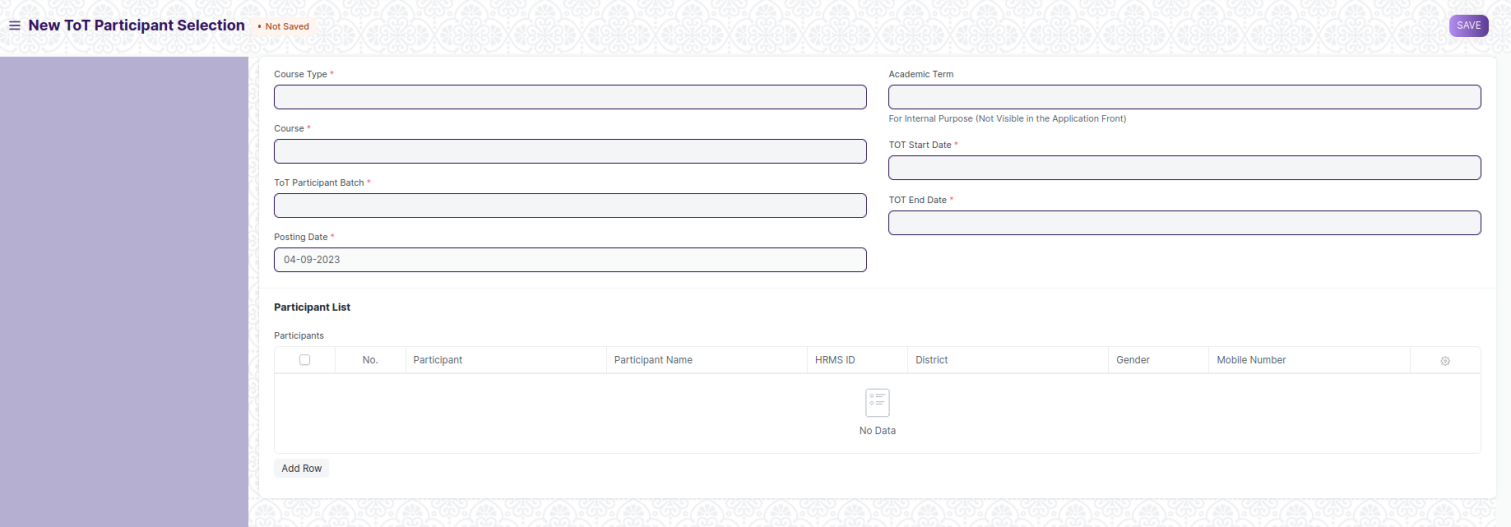


Figure 3 ToT Participant Selection

## ToT Participant Enrollment

Description: This screen is provided to the user for the enrollment of selected participants into their respective courses.

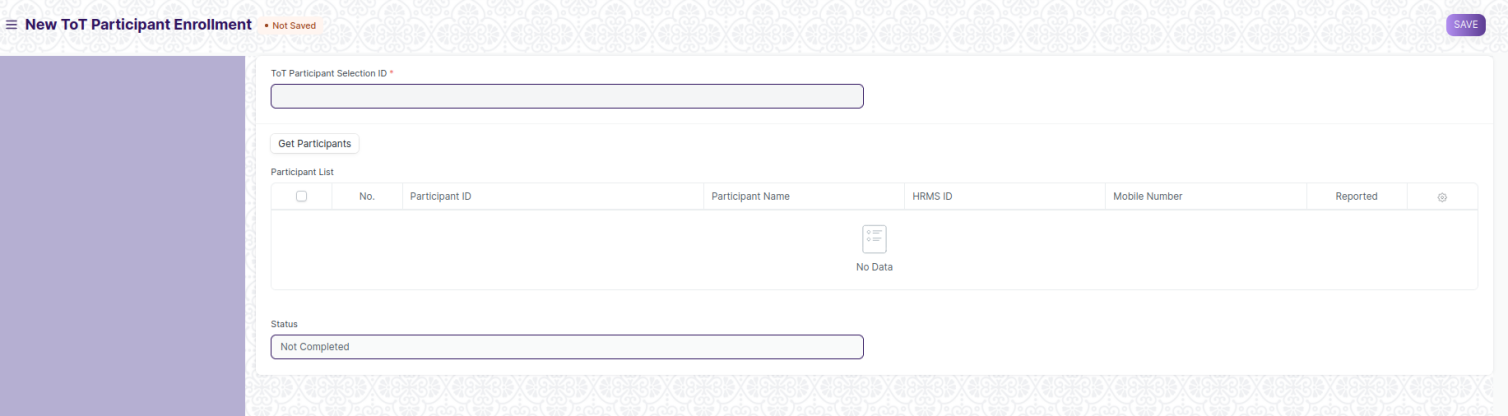
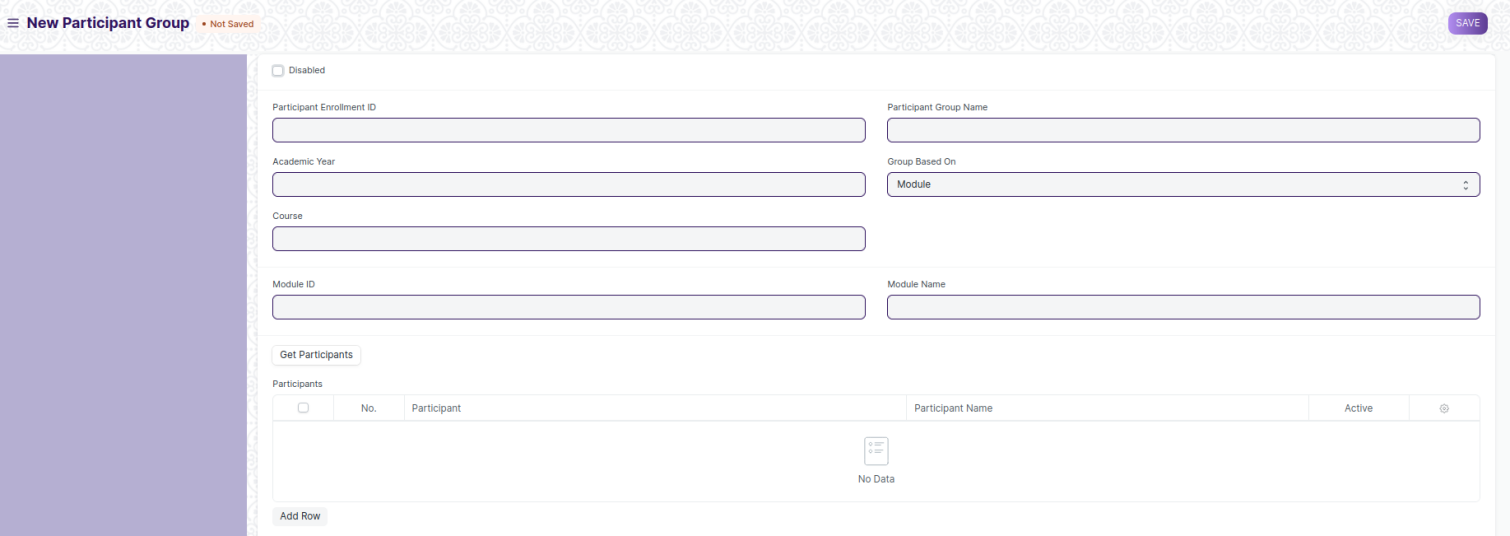


Figure 4 ToT Participant Enrollment

## Participant Group

Description: This screen is provided to the user for the creation of participant groups, consisting of ToT Participants.



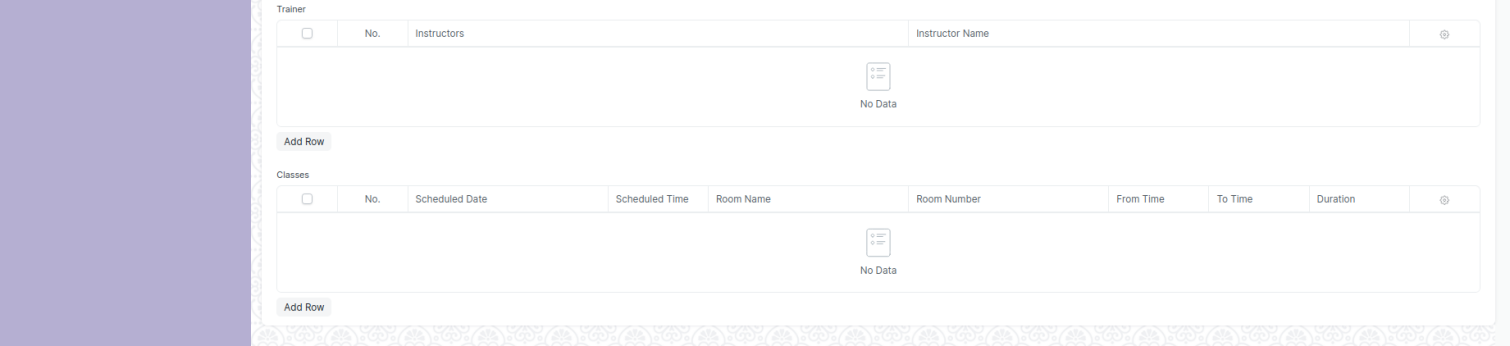
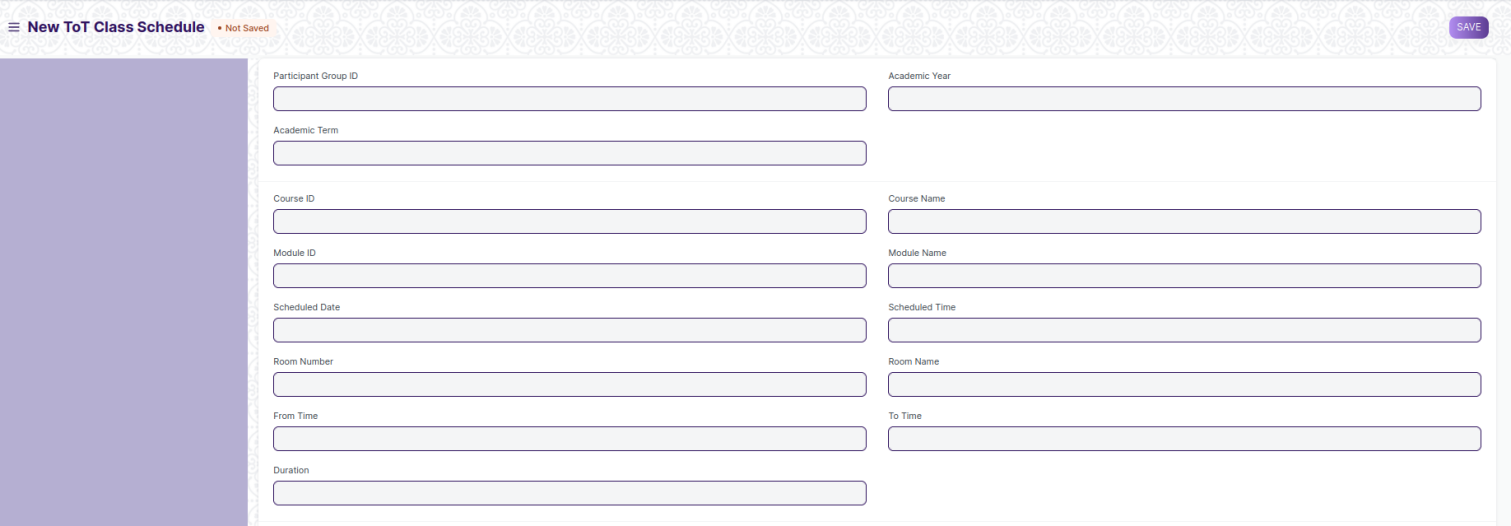


Figure 5 Participant Group

## ToT Class Schedule



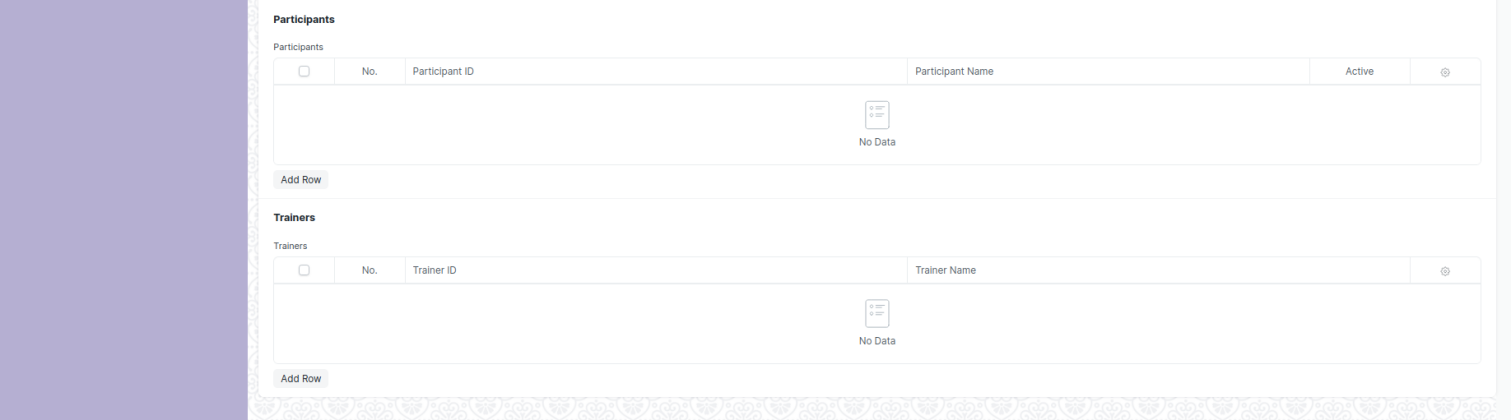


Figure 6 ToT Class Schedule

## ToT Participant Attendance

Description: This screen is provided to the user for recording attendance of participants individually.



Figure 7 ToT Participant Attendance

## Participant Attendance Tool

Description: This screen is provided to the user for recording the attendance of participants in a participant group in bulk.



Figure 8 Participant Attendance Tool

## Assignment

Description: This screen is provided to the user for assiging assignments to the participants of a tot event.

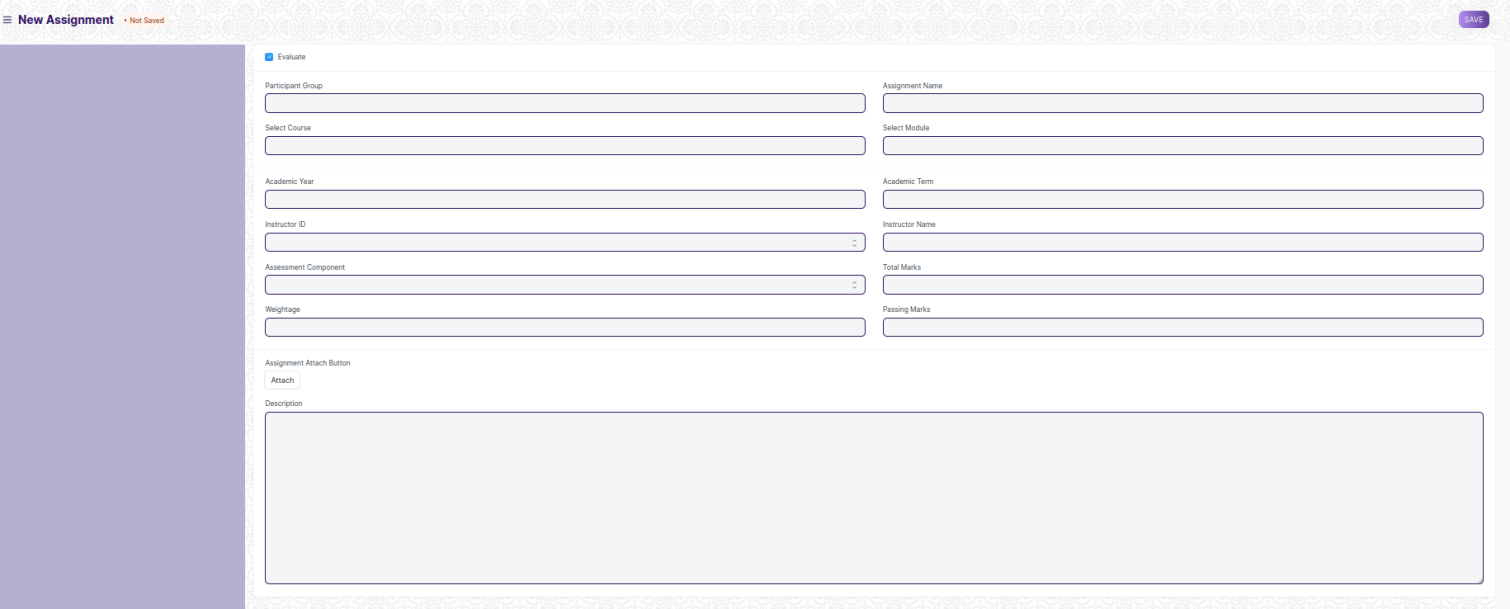


Figure 9 Assignment

## Assignment Upload

Description: This screen is provided to the user for uploading their responses to an assigned assignment.

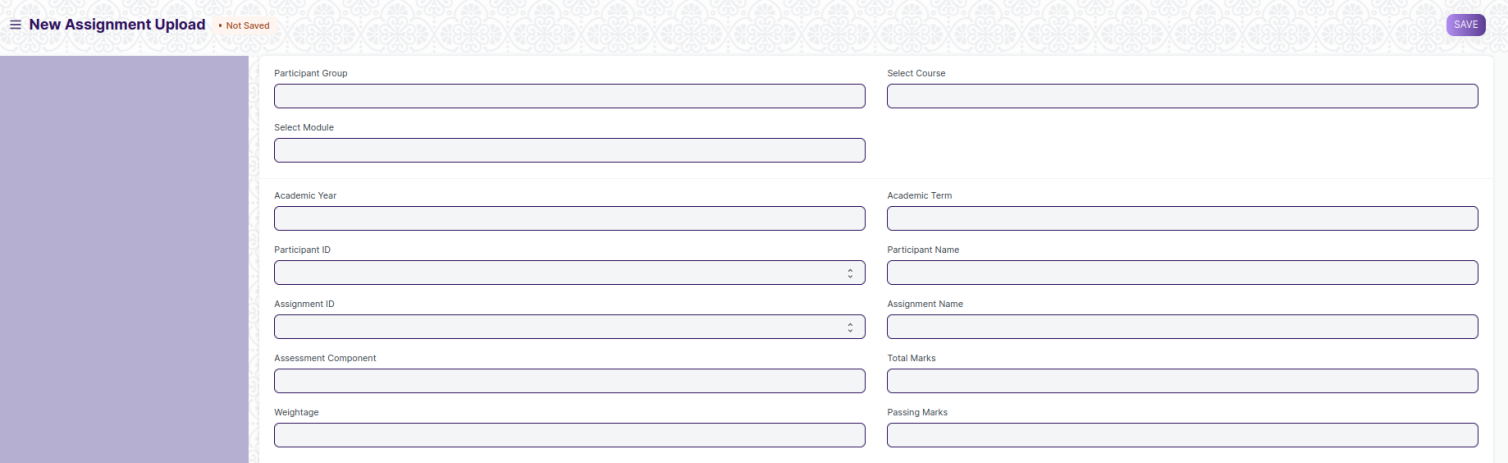
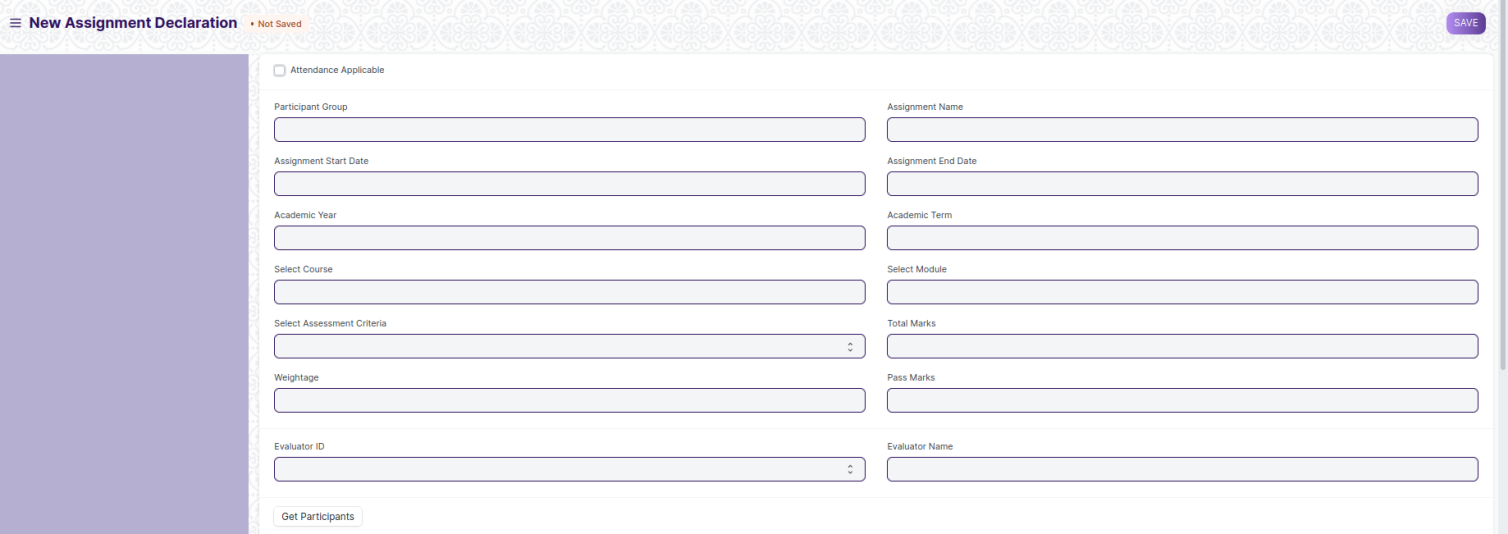




Figure10 Assignment Upload

## Assignment Declaration

Description: This screen is provided to the user for declaring assignments to the participants of a ToT event



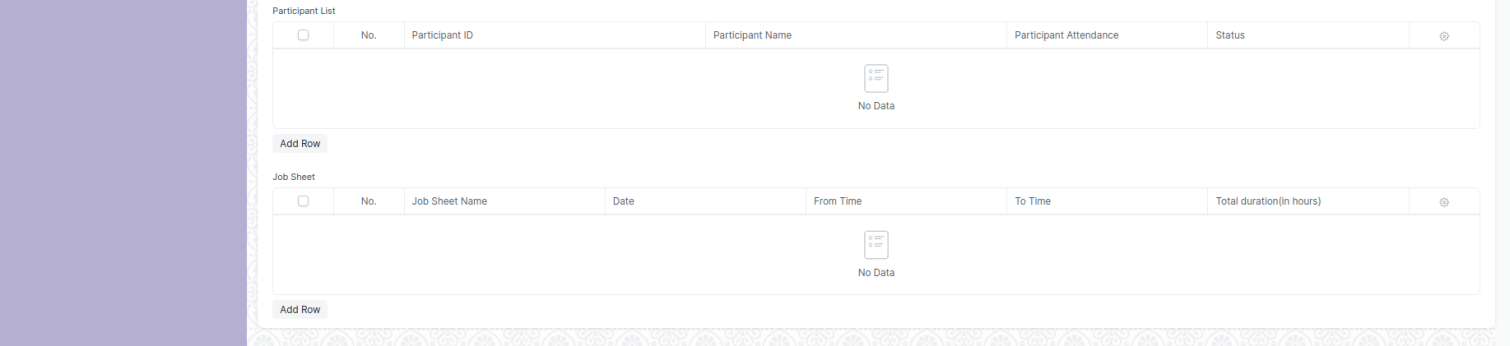


Figure 11 Assignment Declaration

## Assignment Evaluation

Description: This screen is provided to the user for evaluating assignments submitted by the participants of a ToT event individually.

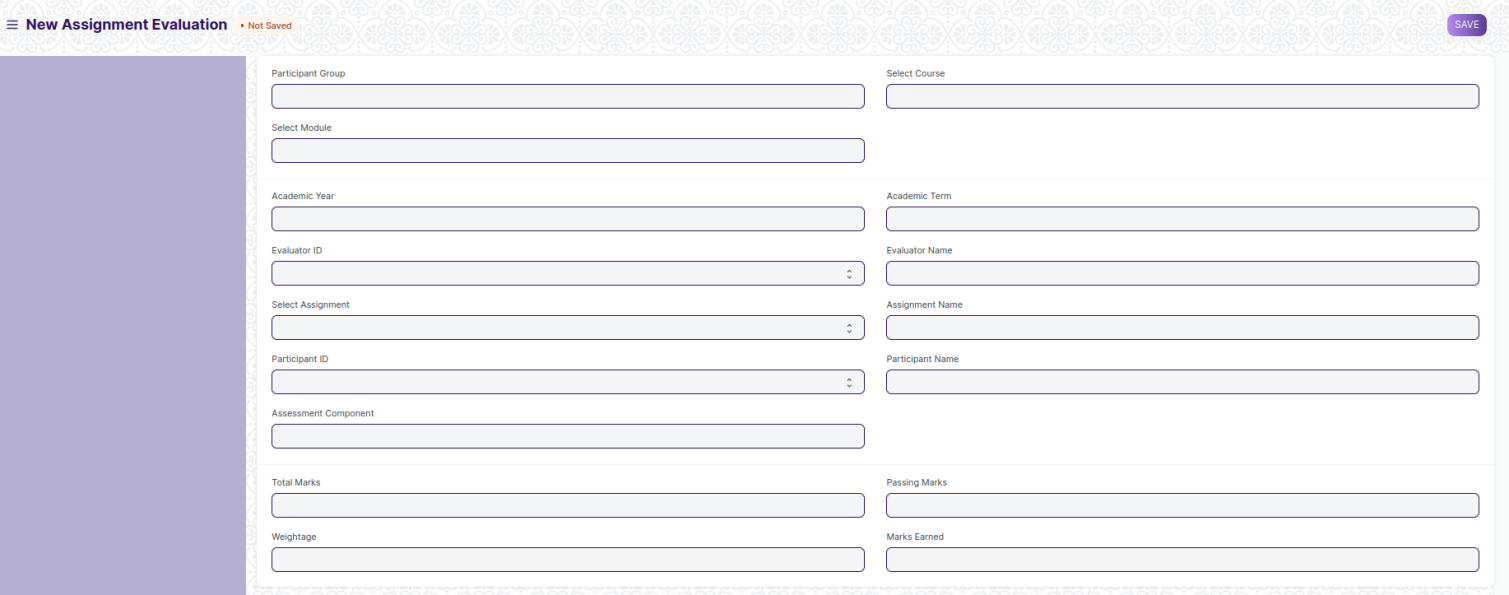
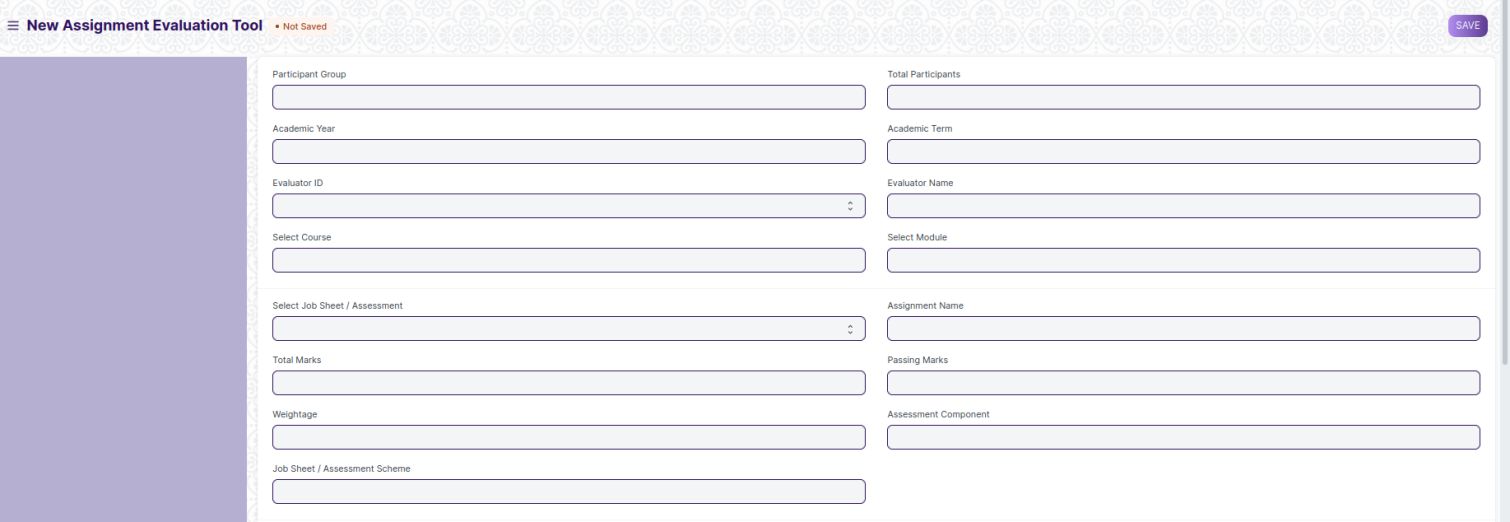


Figure 12 Assignment Evaluation

## Assignment Evaluation Tool

Description: This screen is to be provide to the user for evaluating assignments submitted by the participants of a ToT event in bulk.



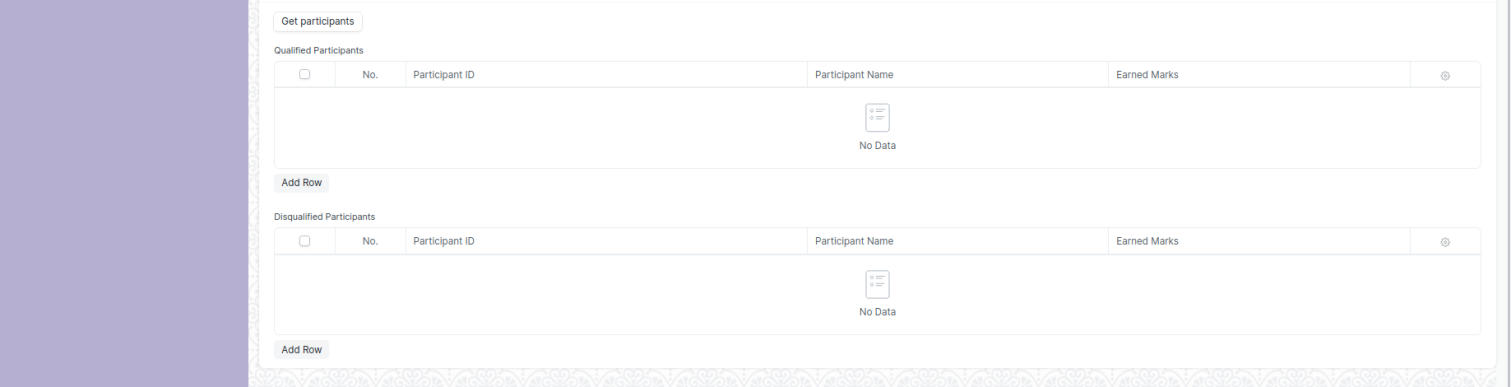
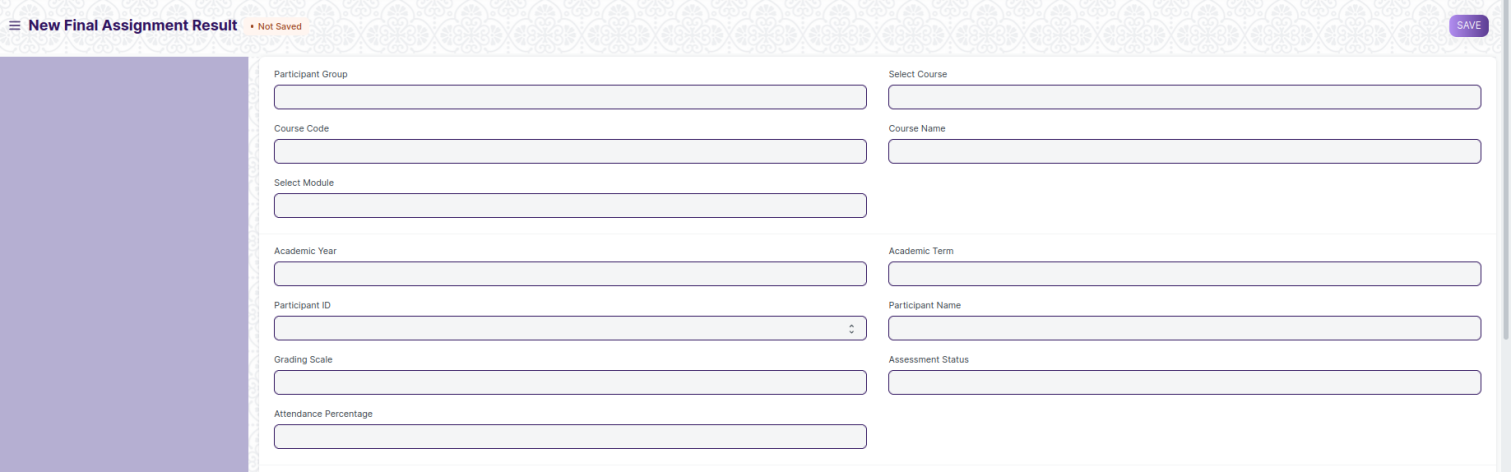


Figure 13 Assignment Evaluation Tool

## Final Assignment Result

Description: This screen is to be provided to the user to generate the overall performance of a ToT Participant in a ToT event and issue the qualified ones their certificates for the event.



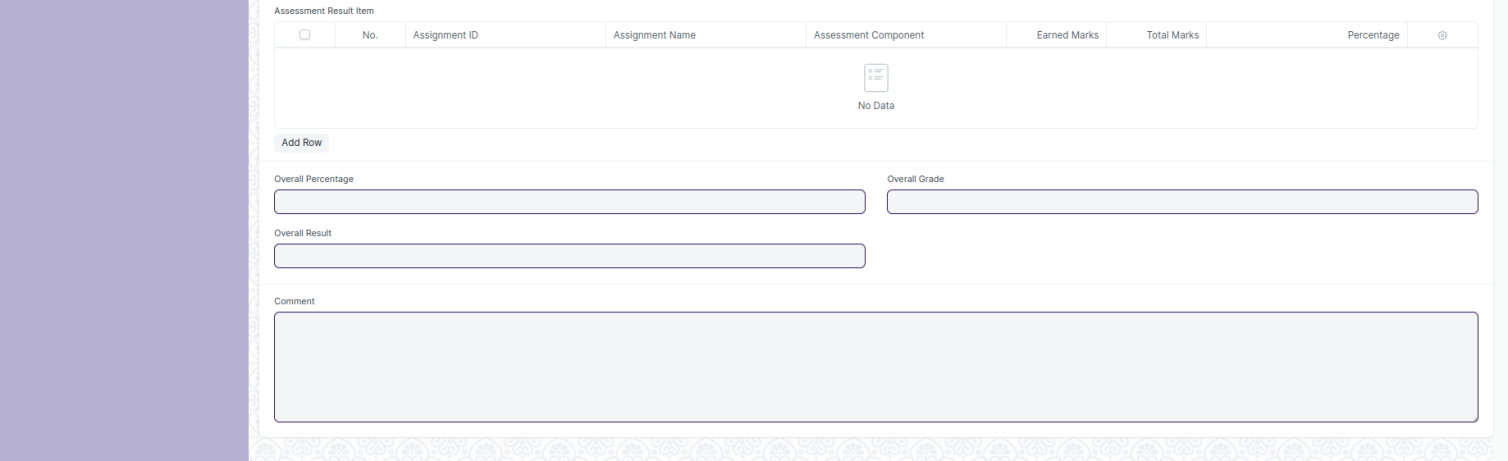


Figure 14 Internship Completion Status

## Final Result Declaration Tool

Description: This screen is provided to the user to generate the overall performance of the participants of a participant group in a ToT event in bulk.

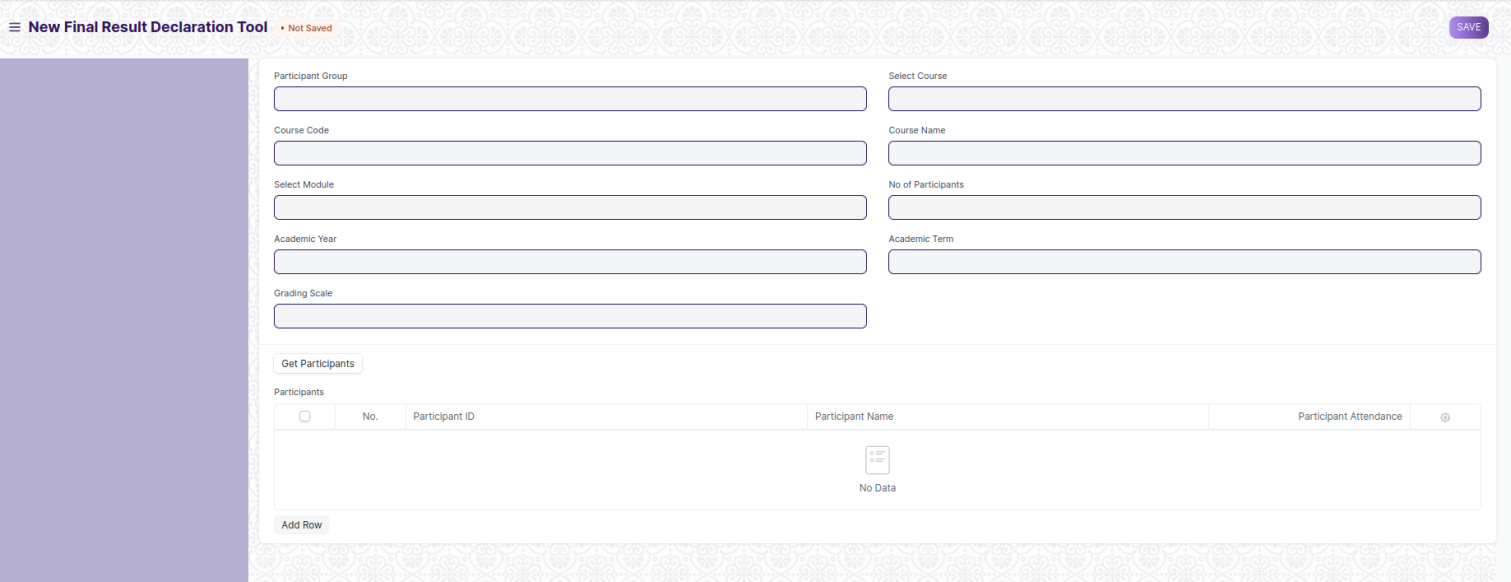


Figure 15 Final Result Declaration Tool

# Definitions and Acronyms

The following table explains the terms and abbreviations used in the document:

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| HLD | High Level Design |
| LLD | Low Level Design |
| API | Application Programming interface |
| NA | Not Applicable |
| ERP | Enterprise Resource Planning |
| HRMS | Human Resource Management System |
| GDPR | General Data Protection Regulation |
| PCI DSS | Payment Card Industry Data Security Standard |

# Deployment Description

* Application Name: Campus Management Application At World Skill Center (WSC)
* Deployment Environment: Production
* Server Information: IP address:117.250.67.19, domain name:erp.worldskillcenter.org, hosting provider: OCAC
* Deployment Date: 13-06-2023

Database Information: MariaDB 10.6.\*, Installed on the same server as the Application