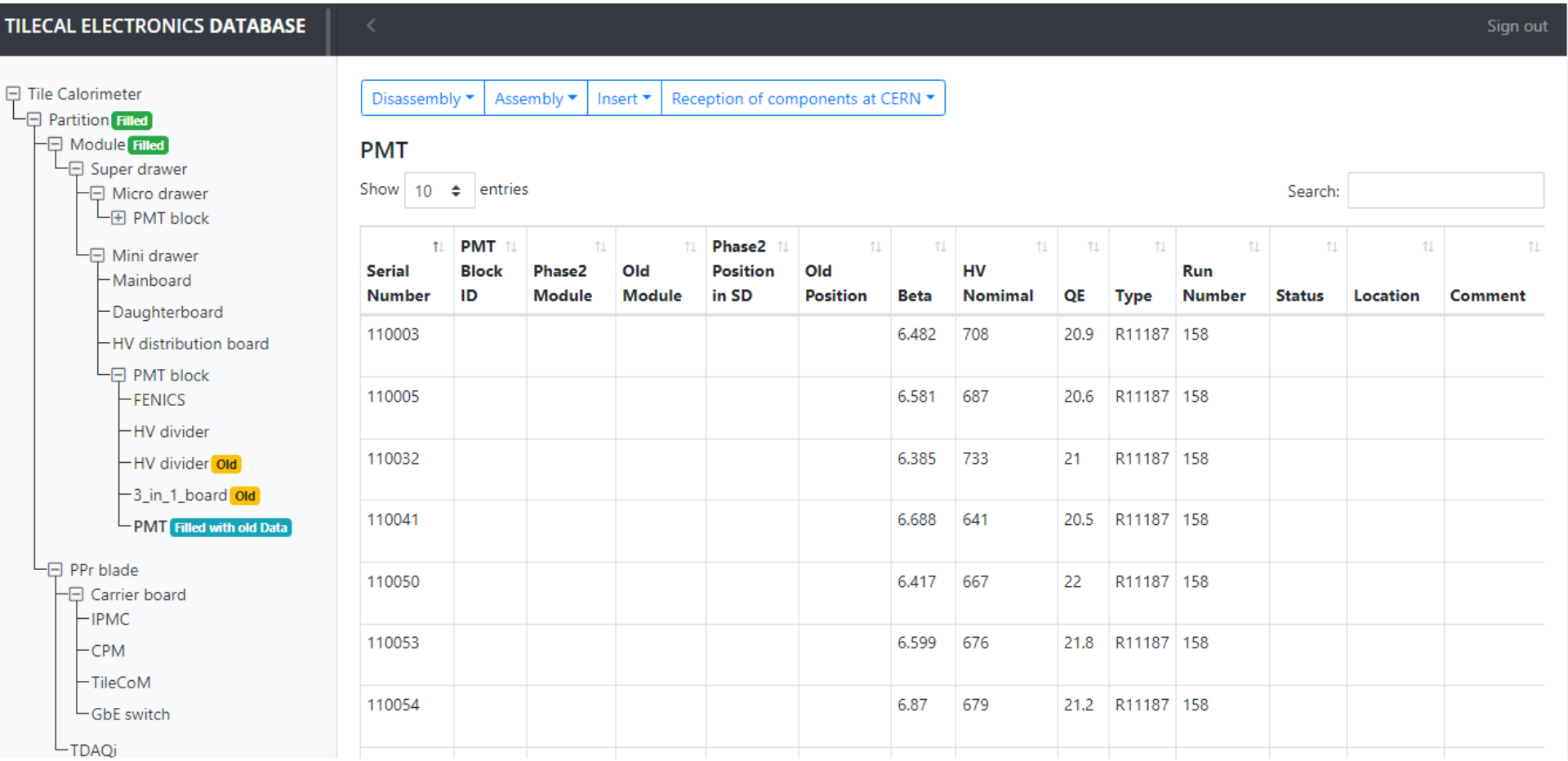
TileCal DB

KVERENCHKHILADZE Irakli
Georgian Technical University

https://indico.cern.ch/event/1040921/

INTRODUCTION



https://tilecal-db.web.cern.ch/#

Working Plan

1st Quarter January-February-March

Project #01: **MySQL DB Tables Development**

1. Creating plan to save component modification histories ✓
2. Setting up backup and recovery plan ✓

Manpower: 1FTE

Total: 4 Weeks

Output: MySQL tables

Project #02: **Development of early Application Functionality**

1. Implementing advanced filter for component properties and add possibility to sort rows ✓
2. Creating priority list to different part of components what is being received at CERN and creating functionality to add such components into the database ✓

Manpower: 1FTE

Total: 6 Weeks

Output: JS/Php Application

Working Plan

2nd Quarter April-May-June

Project #03: **Development of Application Functionality & User Interface**

1. UI: Creating Input Controls (buttons, text fields, check-boxes, radio buttons, drop-down lists, list boxes, toggles, date fields, tables for database records, and input forms for Create, Update and Disassembly/Assembly/Reception of components at CERN processes. ✓
2. UI: Creating Informational Components (icons, progress bar, notifications, message boxes, modal windows) ✗
3. Writing modules for CRUD (Create, Read, Update, Delete) operations for all components. ✗
4. Writing modules for composite component Disassembly/Assembly/Update processes. ✗

Manpower: 1FTE

Total: 13 Weeks

Output: JS/Php Application

Project #1 status

Creating a plan to save component modification histories

Database structure fully supports the feature to save every component modification history, this is done on the server side via MySQL triggers and there is no need for user/web application interaction for the task.

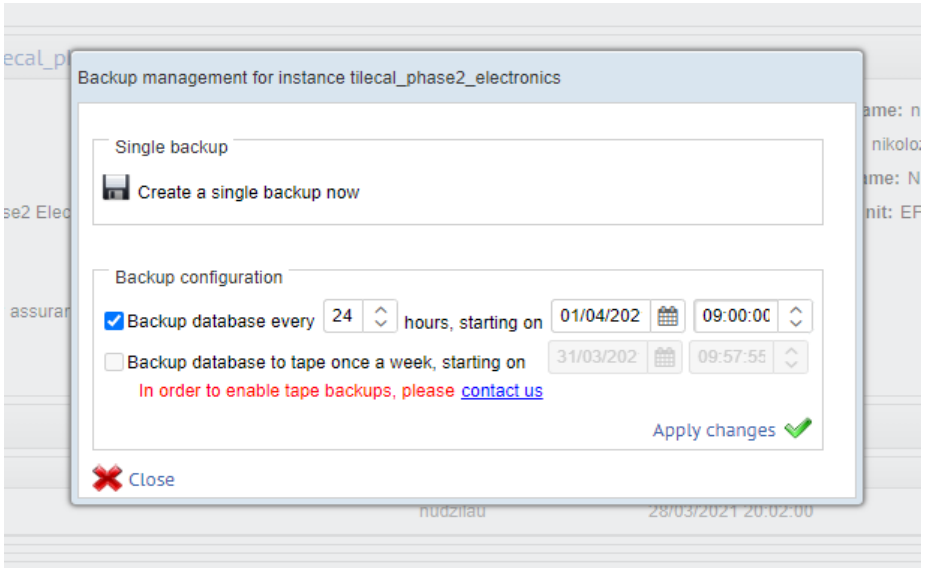
MySQL Trigger Example for PMT table

```
811  CREATE TRIGGER `pmt_hist` AFTER UPDATE ON `pmt`
812   FOR EACH ROW
813      INSERT INTO tilecal_electronics_hist.pmt_hist
814      (serial_number, id_pmt_block, module,
815      module_in_legacy_tilecal, pos_in_legacy_tilecal,
816      pos_in_phase2_tilecal, beta, HV_nominal,
817      QE, pmt_type, run_number, component_status,
818      current_location, remark, start_date, end_date,
819      created_by, changed_by)
820      VALUES (old.serial_number, old.id_pmt_block, old.module,
821      old.module_in_legacy_tilecal, old.pos_in_legacy_tilecal,
822      old.pos_in_phase2_tilecal, old.beta, old.HV_nominal, old.QE,
823      old.pmt_type, old.run_number, old.component_status,
824      old.current_location, old.remark, old.updated_at,
825      NOW(), old.created_by, old.updated_by);
```

Project #1 status

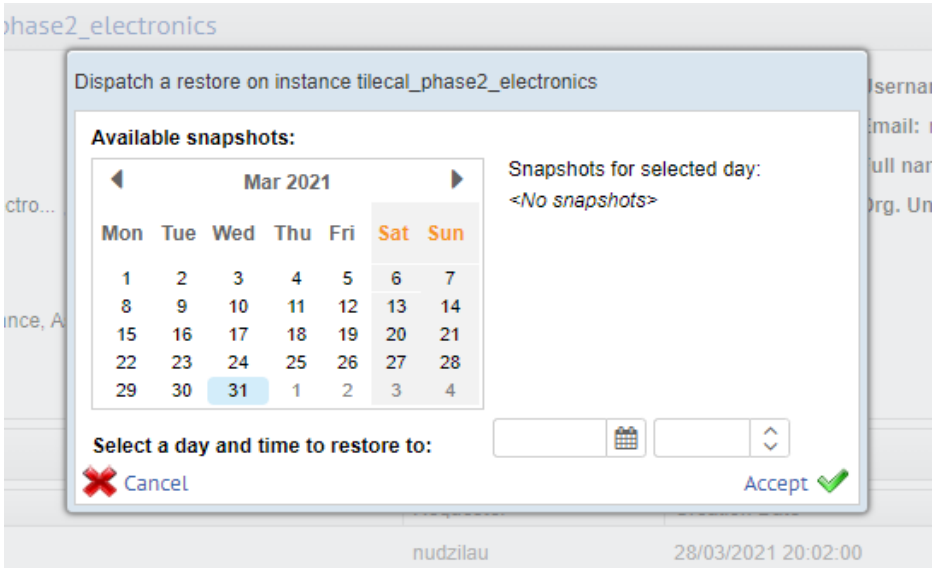
Setting up a backup and recovery plan

Backup:



We use “CERN DATABASE ON DEMAND SERVICE” which gives the feature to set up automatic backups. Below is the backup configuration example.

Restore:

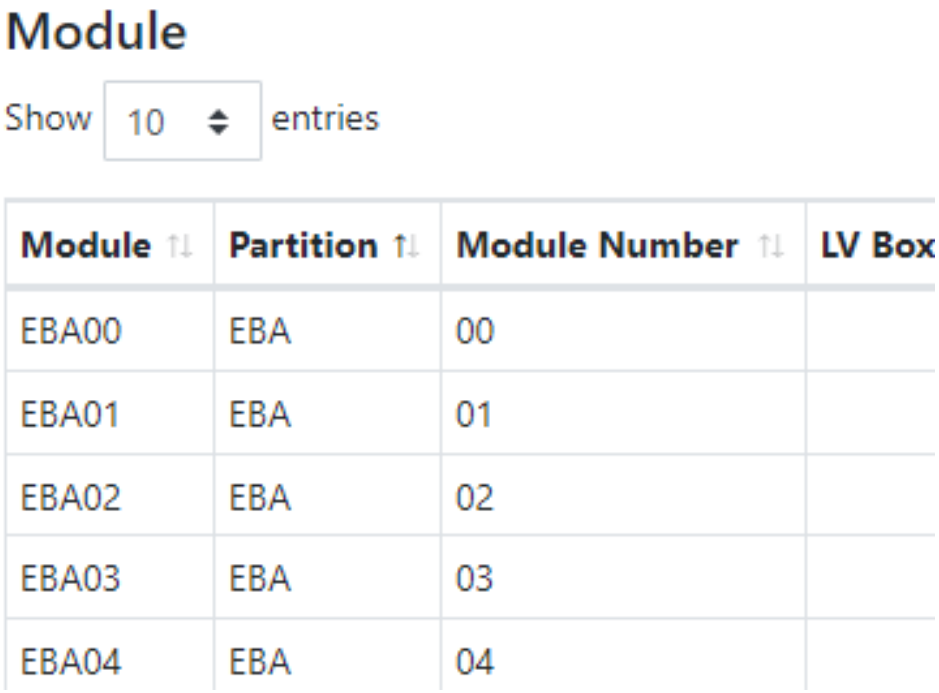


Restore is also managed through DBOD Service, by selecting the date and time to restore.

Project #2 status

Implementing advanced filters for component properties and add possibility to sort rows

1. Possibility to **sort rows** (ASC, DESC)
2. **Normal filter**, “possibility to filter values of components by specifying exact value” e.g. 110003 or “LBA01”



Project #2 status

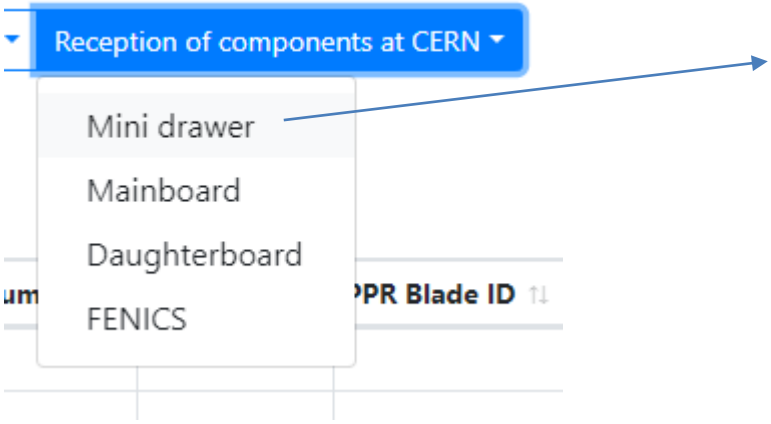
Creating priority lists to different parts of components that are being received at CERN and creating functionality to add such components into the database

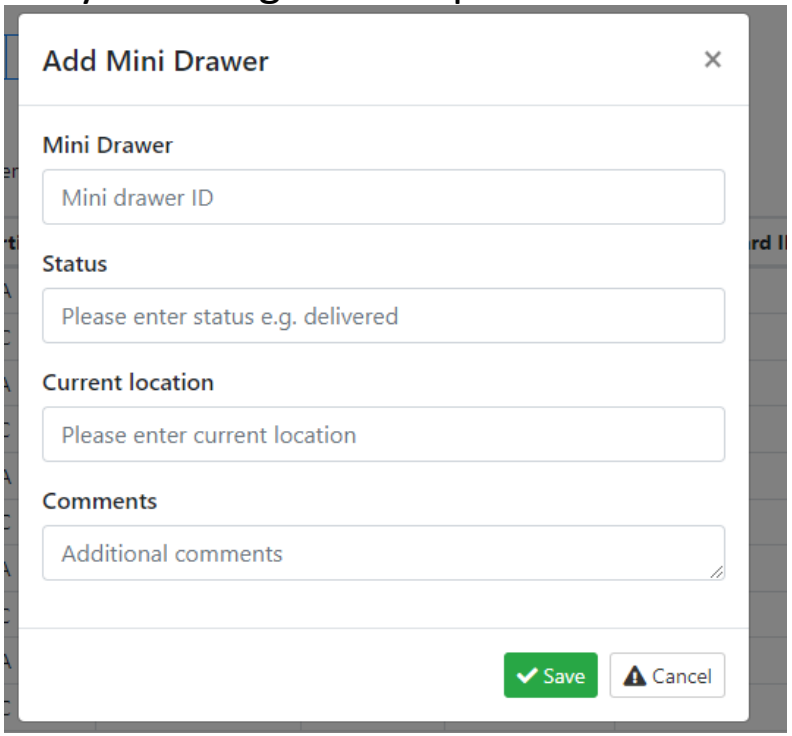
The procedure from application is the following:

Choose the component to add -> Fill the form -> Click Save.

The Fields:

1. **Component ID** (Required) (Can be entered by scanning the component barcode or entering it manually)
2. **Status** (e.g. delivered, etc.)
3. **Current Location** (Storage #, etc..)
4. **Additional Comments**.





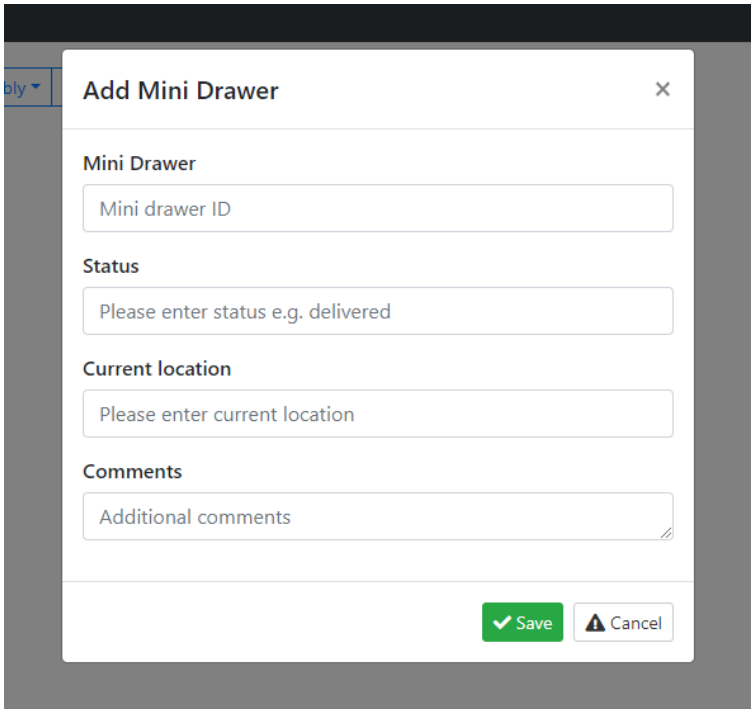
Future Plans

2.1 UI: Creating Informational Components (icons, progress bar, notifications, message boxes, modal windows)

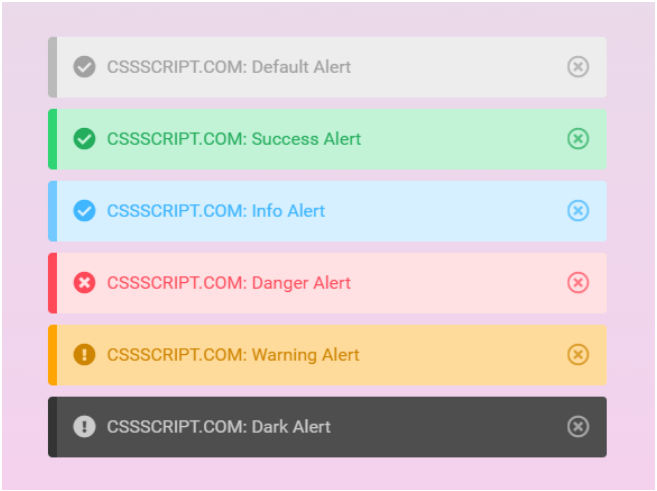
Progress bar

Notification

Modal Window



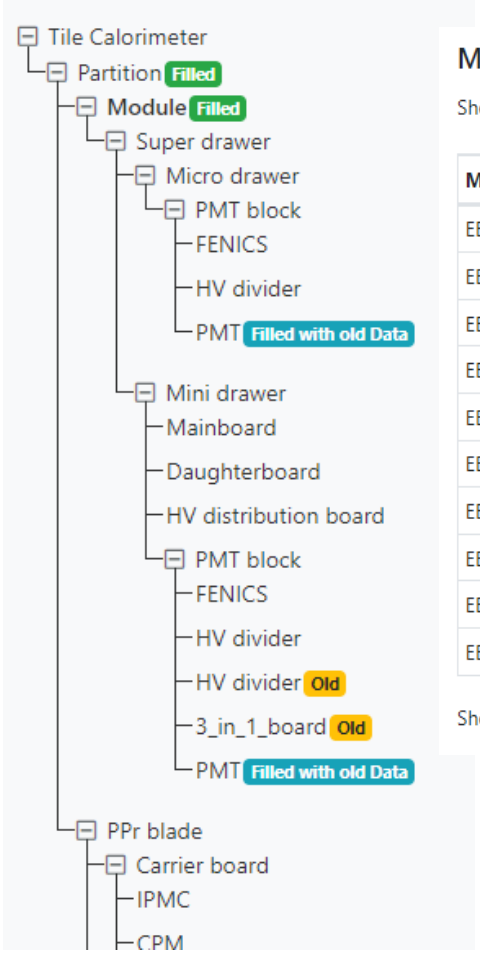
Message box examples



Future Plans

2.2 Writing modules for CRUD (Create, Read, Update, Delete) operations for all components

Components



Module

Update icon

Show 10 entries

Search:

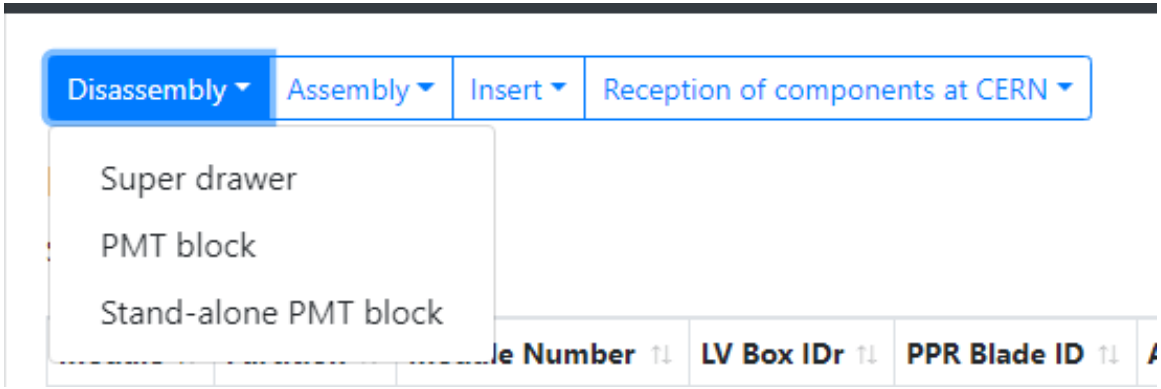
Module	Partition	Module Number	LV Box IDr	PPR Blade ID	Auxiliary Board ID	HV Regulation Board ID	Comment	Action
EBA00	EBA	00						
EBA01	EBA	01						
EBA02	EBA	02						
EBA03	EBA	03						
EBA04	EBA	04						
EBA05	EBA	05						
EBA06	EBA	06						
EBA07	EBA	07						
EBA08	EBA	08						
EBA09	EBA	09						

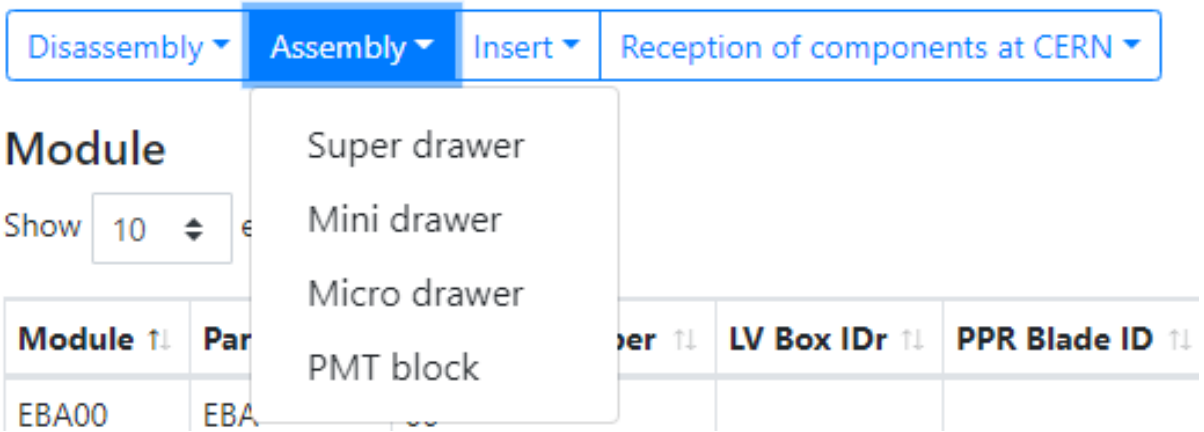
Showing 1 to 10 of 264 entries

Previous 1 2 3 4 5 ... 27 Next

Future Plans

2.3. Writing modules for composite component Disassembly/Assembly/Update processes





Future Plans

3nd Quarter July-August -September

Project #04: **Development of main functionality and connecting them to UI elements**

1. Writing module of Barcode recognition based on barcode structure
2. Writing module for bringing either existing record for scanned component if it is already in the DB, or bringing an interface for that specific component that proposed to create a corresponding entry
3. Connecting components tree to a database
4. Writing module to give assistance during the process of inserting/updating data in the database (providing hints, giving warnings in case of inaccuracy and in case of any type of system error)
5. Creating modules for additional requirements. (Recognize super drawer structure based on partition and module; Check compatibility between the PMT block and the slot which it is being inserted in; Recognize restricted positions for PMT blocks in Super drawer, etc.)

Manpower: 1FTE

Total: 13 Weeks

Output: JS/Php Application

Deadline&Workaround

We are late in deadline and our main priority is to catch up to the working plan.

Workaround :

1. 2nd quarter finish time : 30 July
2. 3rd quarter finish time : 30 September

Thanks for your Attention!

Comments are Welcome

irakli.kverenchkhiladze@cern.ch