



INSTALLATION AND COMMISSIONING DATABASE STATUS

SHARMAZANASHVILI Alexander
Georgian Technical University

UDZILAURI Nikoloz
Georgian Technical University

I.	Implementing a database on a specific RDBMS (MySQL)	4m
	1. Creating physical entities	
	2. Defining entities (adding properties)	
	3. Building relationships	
	4. Creating Triggers	
	5. Creating Statements	
	6. Creating plan to save component modification histories	
	7. Setting up backup and recovery plan	
	8. Setting up security	

II.	Prefilling the database with old PMT's spreadsheet	3m
	1. Developing application program to parse old PMT's spreadsheet	
	2. Filling new Database with parsed data	

III.	Creating the User Interface	2m
	1. UI: Drawing/planning UI based on application functionality	
	2. UI: Creating page layout (Building basic UI structure)	
	3. UI: Creating containers (Creating containers for different components of UI)	
	4. UI: Creating Navigational Components (breadcrumb, search (filter) field, pagination, Tile Cal electronics components structure tree)	
	5. UI: Creating Input Controls (buttons, text fields, checkboxes, radio buttons, dropdown lists, list boxes, toggles, date fields, tables for database records, Input forms of disassembly/assembly/ /update/Reception of components at CERN processes)	
	6. UI: Creating Informational Components (icons, progress bar, notifications, message boxes, modal windows)	

IV.	Developing of the application externals	2m
	1. Creating the Gitlab repository	
	2. Setting up connectivity with Database	
	3. Setting up authentication and application security	
	4. Functionality testing, Usability testing, Interface testing, Compatibility testing, Performance testing, Security testing	

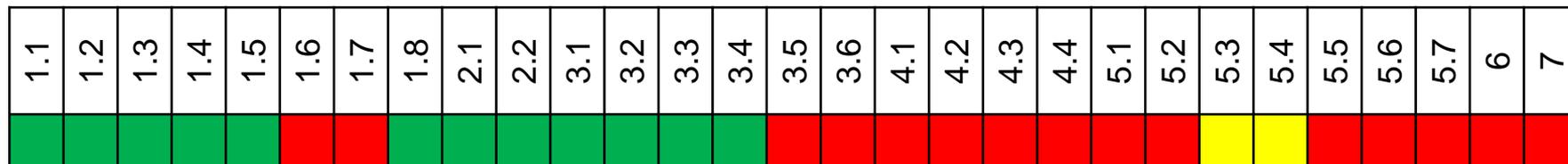
V.	Development of main functionality and connecting them to UI elements	5m
	1. Writing module of Barcode recognition based on barcode structure	
	2. Writing module for bringing either existing record for scanned component if it is already in the DB, or bringing an interface for that specific component that proposed to create a corresponding entry	
	3. Connecting components tree to a database	
	4. Writing module for search and filter of database components and component properties	
	5. Creating modules for CRUD (Create, Read, Update, Delete) operations and connecting them to UI	

V.	Development of main functionality and connecting them to UI elements 5m
	6. 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.)
	7. 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)

VI.	Development of technical documentation of application code and database structure	2m
-----	---	----

VII.	Creating user manual providing troubleshooting, with the explanation of conflict cases and possible ways for the solution	2m
------	---	----

Status Bar: 02-12-202



Releases: 02-12-202

	R01:	3.1; 3.2; 3.3; 3.4
	R02:	5.3; 5.4
	R03:	3.5; 3.6
	R04:	5.5
	R05:	5.1; 5.2
	R06:	5.6; 5.7
	R07:	4.2; 4.3; 4.4

- You can reach the application here

<https://tilecal-electronics-db.cadcamge.ch/pmt>

TILECAL ELECTRONICS DATABASE

Disassembly

Assembly

Reception of components at CERN

CREATE/UPDATE

Database

- Tile Calorimeter
 - Partition Filled
 - Module Filled
 - Super drawer
 - Micro drawer
 - PMT block
 - FENICS
 - HV divider
 - PMT
 - Mini drawer
 - Mainboard
 - Daughterboard
 - HV distribution board
 - PMT block
 - FENICS
 - HV divider
 - HV divider Old
 - 3_in_1_board Old
 - PMT Filled partly

PMT Add PMT

PMT serial_number	PMT Block ID	Module	Pos in legacy tilecal	Pos in phase2 tilecal	Beta	HV nominal	QE	Type	Status	Current I
<input type="text" value="PMT serial_numbe"/>	<input type="text" value="PMT Block I"/>	<input type="text" value="Modt"/>	<input type="text" value="Pos in legacy tileca"/>	<input type="text" value="Pos in phase2 tileca"/>	<input type="text" value="Be"/>	<input type="text" value="HV nomin"/>	<input type="text" value="Q"/>	<input type="text" value="Type"/>	<input type="text" value="Stat"/>	<input type="text" value="Current"/>
110003					6.482	708	20.9	R11187		
110005					6.581	687	20.6	R11187		
110032					6.385	733	21	R11187		
110041					6.688	641	20.5	R11187		
110050					6.417	667	22	R11187		
110053					6.599	676	21.8	R11187		
110054					6.87	679	21.2	R11187		
110055					6.451	678	22.3	R11187		
110056					6.796	660	22.7	R11187		
110058					6.599	652	22.2	R11187		
110062					6.759	686	20.5	R11187		
110064					6.663	714	20.4	R11187		
110068					6.786	667	22.3	R11187		
110069					6.818	662	21.8	R11187		
110074					6.694	675	22.6	R11187		
110077					6.737	670	19.8	R11187		
110079					6.665	644	21.1	R11187		
110080					6.655	694	21.6	R11187		
110090					6.76	657	23	R11187		
110092					6.758	666	22.2	R11187		
110098					6.445	674	21.7	R11187		
110100					5.966	732	21.6	R11187		

- Please send us the feedback. It is very important for the development life cycle

Thanks for the attention,

Comments are welcome

Lasha.sharmazanashvili@cern.ch

nikoloz.udzilauri@cern.ch