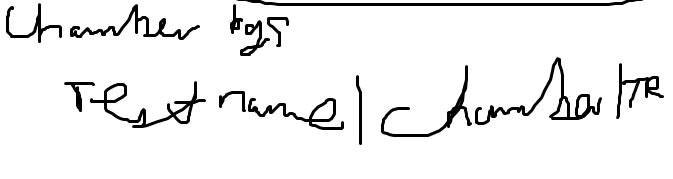
**Web pages:**

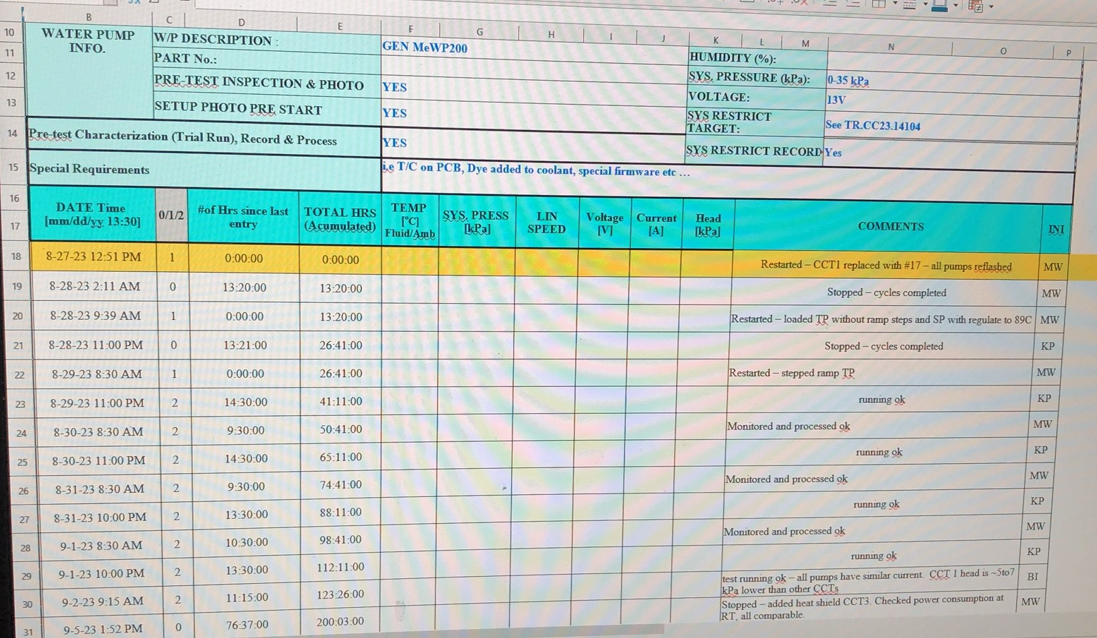
1. Show all the information (columns) of a table. Allow users to add, change, and remove rows. Also, users will be able to filter, sort, search for items in these tables. These web pages will be for the following tables:

* All tables except Test, Chamber Logs, Chamber Log Info.

1. For **Chamber Logs, Chamber Lo**g Info, we will have a “Chamber Logs” web page from which you can initially add a “Chamber Log Info” row as below



Then, when we click on a chamber log, we will see a screen like the current log file,



Where we can add logs that will be rows in the “Chamber Log” table, and fill in additional information for the row on “Chamber Log Info” table.

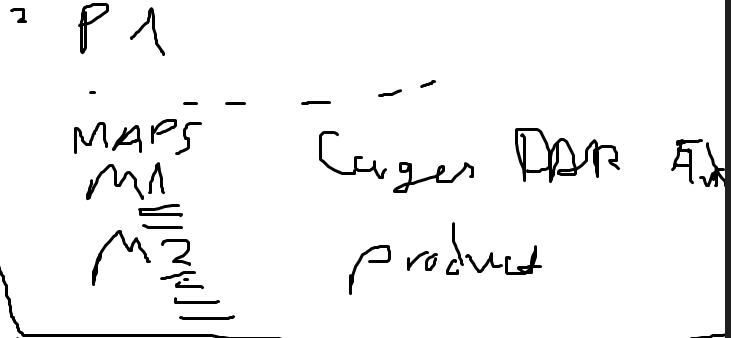
1. For the **Test** web page, we will display the following columns in addition to the column in the “Test” table (we will add them from other tables through queries):

* Product family
* Program name
* Leg name
* TR of leg

\*Need to add priority in the test table.

This web page will still allow users to add, edit, and remove test. But they cannot add in new rows of additional columns.

1. We will also have an additional web page for **Programs.** We access this page through hyperlink from the Program table page when the user clicks on a Program. This will bring us to a page with the program name as the title, and it will display everything from the program table, along with its test maps, and tests, structured in a tree format. We will also show DARs, Chambers, Cages,… from relationship tables that are compatible with it. As shown below:



1. **Log in** page.
2. **User role** page.

**Features:**

* Edits, removals from a table will be prompted with a confirmation message.
* Have a calendar when adding Test rows to visualize availability.
* When there’s a conflict in test scheduling times, display an error.
* Validate that data to be added in tables are the right datatype, length, and primary keys are unique.

**Next steps:**

1. Validate database. - OK
2. Establish list of all web pages that will be in web app - Ok
3. Agree on structure, features of important web pages (e.g. test), and smaller web pages (identical web pages). - Ok
4. Determine queries needed.
5. Meeting with team.
6. Program web pages.
7. Organize how web pages link to each other (creating menu for example).