

Numbine

“An Application for Optimized Configuration Testing”

Web Page Design Document

Rajasekaran Rajagopal

Version 1.0

Date: 5/31/2021

Project Sponsor: Nir Harel



DRAFT

Version history

Ver. No.	Date	Description	Name	Note
1.0	5/12/2021	Initial Draft: Main role-based login Page, Admin Page, Admin Access permission page, User Config. Generate Page (Parameter file, On-demand), User Dashboard page, Admin Dashboard Page, and Admin Chart Page	Raj	
1.1	6/7/2021	Added: User product group, delete user alert, "Save to Database" with "Commit" alert box with Release name, product version, test cycle, group, download input parameter file, and delete config file alert for both user and admin.	Raj	Update after Reviewed with Wassim
1.2	6/8/2021	Worked database/table	Raj	
1.3	6/16/2021	Updated user & admin dashboard with the search option based on product and release, added product field display. Admin "Access" page with update button and add button only for insert user row. On user config generator page added "Pre-defined configuration parameters option with product, release, and version pulldown box	Raj	
1.4	6/17/2021	Working on configurator pages		
15	6/20/2021	making the doc simpler to the course students		

Table of Content

Version history	2
Table of Content	3
General	5
Purpose	5
Overview DELL Primary Storage Products	5
Actors	6
Wireframes & functionality	7
1). Main Page	7
2) Admin	8
1) Admin Menu	8
2) Admin home page	8
3) Admin Dashboard Page	9
3) Admin Access Page	10
Admin Access Page - New/Update User	11
3) User	11
1) User Menu	11
2) User Dashboard page	11
3) User Config Generator page	12
4) Configurator	14
1) Configurator Menu	14
2) Configurator Products page	15
3) Configurator User Mapping page	16
4) Configurator Releases Page	18
5) Configurator Test Configuration Page	20
6) Configurator Product Release Mapping Page	22
Technical notes	25
Numbine Application ER Diagram	25
Sample: What is DELLEMC PowerStore?	25

Tables content	27
Table Name: users	27
Table Name: Products	27
Table Name: users_products	28
Table Name: products_releases	28
Table Name: prod_rel_2_tsc_values (Product Release Test Configuration Values)	29
Table Name: test_configurations	29

General

Purpose

Dell products have complex configurations that include many parameters and values, there are several kinds of restrictions on those parameters and their values. This complexity is manifested in complex testing environments and thousands of configuration options that require testing.

The goals of this system is to:

1. Document the configuration options for each product release
2. Minimize the number of configurations to test by connecting the system to a tool that will create the minimum set of configurations that require testing, with priority
3. Knowledge conservation of the planned testing environments

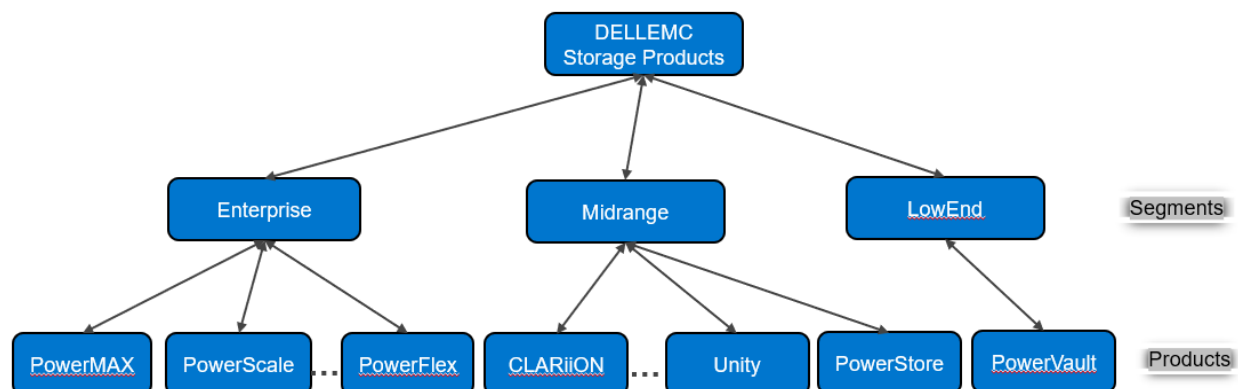
Overview DELL Primary Storage Products

There are number of primary storage products developed that are serving DELL customers, a few of them are in development, and a few are in incubation stage. DELLEMC primary storage product focuses on enterprise(eg: Citi Bank, Walmart etc.),mid and low end customers from different types of industries.

Note: Not listed all the products here, to know in detail you can visit the below web url for more products.

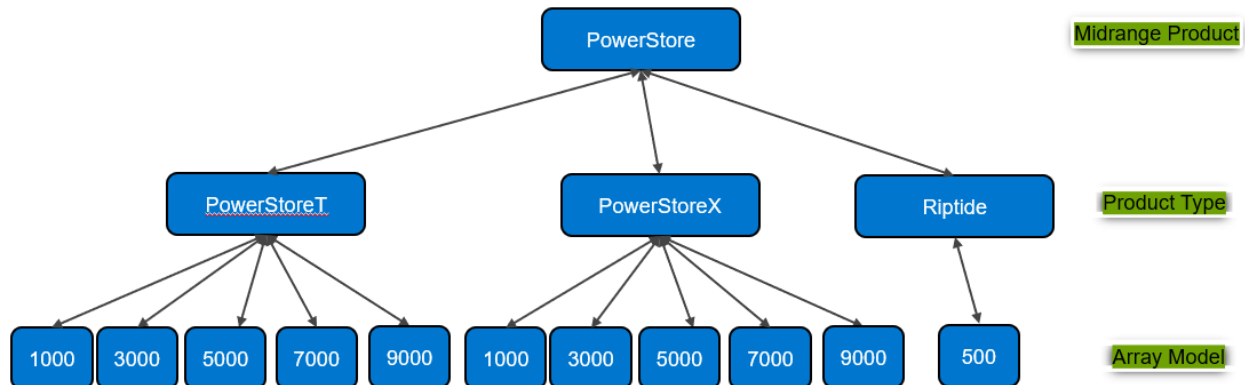
<https://www.delltechnologies.com/en-us/storage/data-storage.htm>

DELLEMC Primary Storage Products

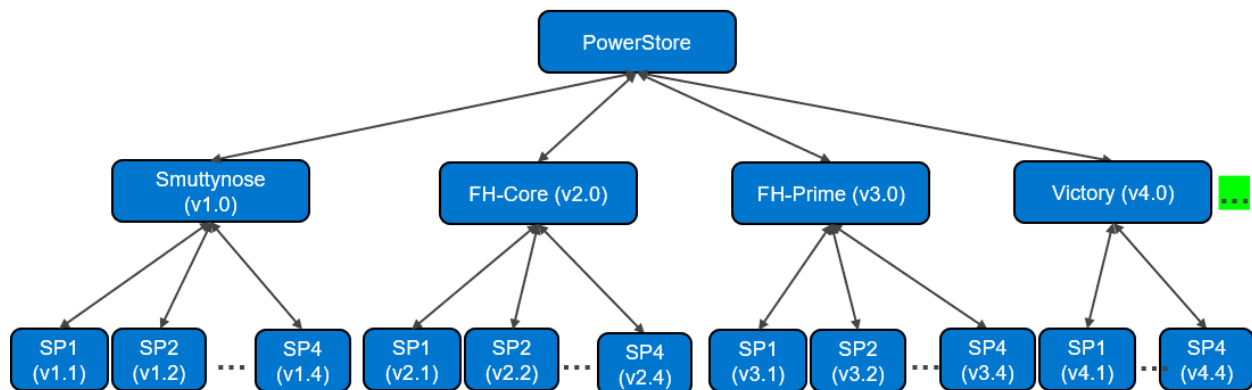


DELLEMC PowerStore is targeted towards midrange and low-end customers. In PowerStore product line we have three product types each product type has five different models as shown below. The model variants defined based on number of CPU/Sockets, memory, Cache, number of drives, total capacity, etc.,

DELLEMC PowerStore Product – Types and Models



DELLEMC PowerStore Product – Releases



eg: SP1 – Service Pack 1

Actors

The system has 3 main actor roles:

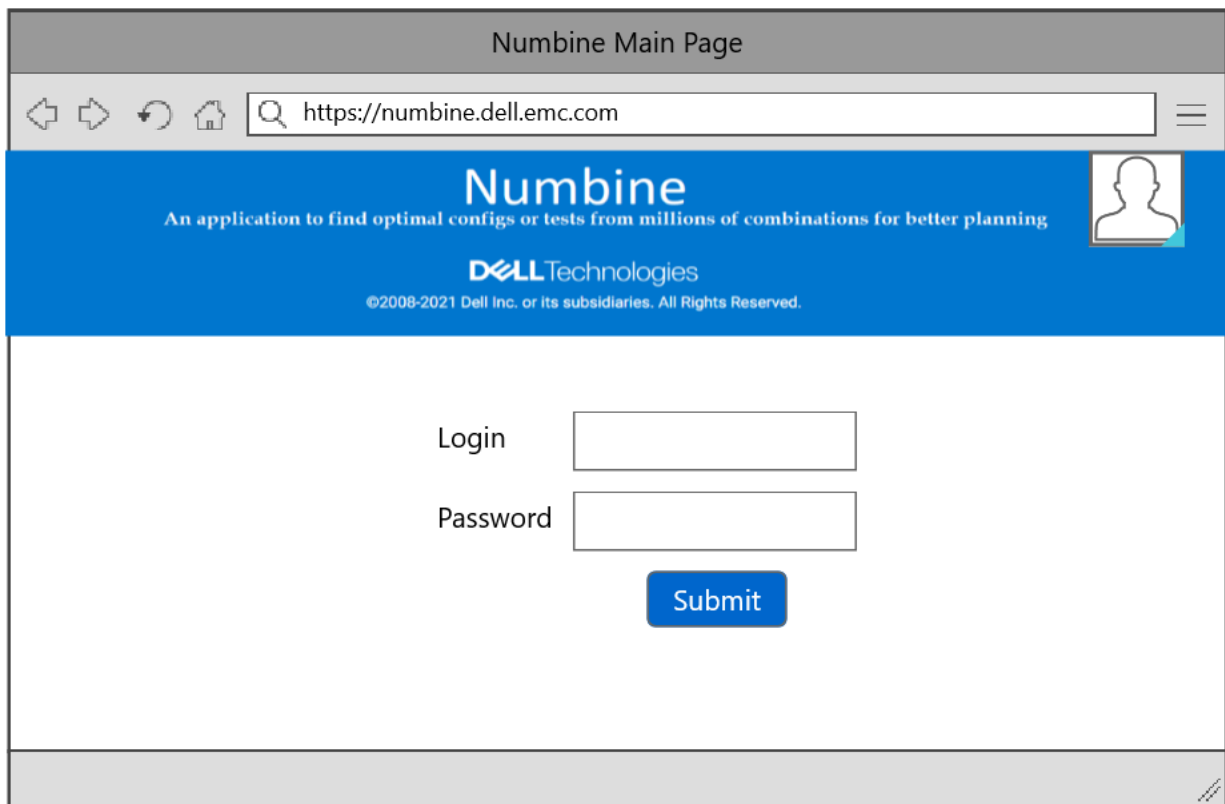
1. Configurator - this role allows the user to CRUD the products, their versions, configuration parameters, values and constraints
2. Admin - this role has 2 main goals:
 - a. allows the user to CRUD users and their roles and permissions
 - b. View and Delete test configuration files stored by the users
3. User - this role is the test user that will use predefined products and versions and will create specific test configurations for specific cycles

A user can have one role only.

Wireframes & functionality

1). Main Page

1. Numbine application main page should support role-based user logins.
2. The “Forgot pwd” button will send the user an email with a new pwd to be used in the Login page.
 - a. in second priority, there will be another form where the user will enter the temp pwd sent to her by email and enter a new pwd to be used
3. Default session time out options (lower priority)



The wireframe shows a web browser window titled "Numbine Main Page". The address bar displays "https://numbine.dell.emc.com". The page header is blue and contains the "Numbine" logo, the tagline "An application to find optimal configs or tests from millions of combinations for better planning", the "DELL Technologies" logo, and a copyright notice "©2008-2021 Dell Inc. or its subsidiaries. All Rights Reserved." A user profile icon is visible in the top right corner of the header. The main content area is white and features a login form with labels "Login" and "Password" next to input fields. A blue "Submit" button is positioned below the password field. The footer is a solid grey bar.

2) Admin

As an Admin user I would like to:

1. manage the users in the system and their roles (CRUD)
2. manage the test config files (RD)

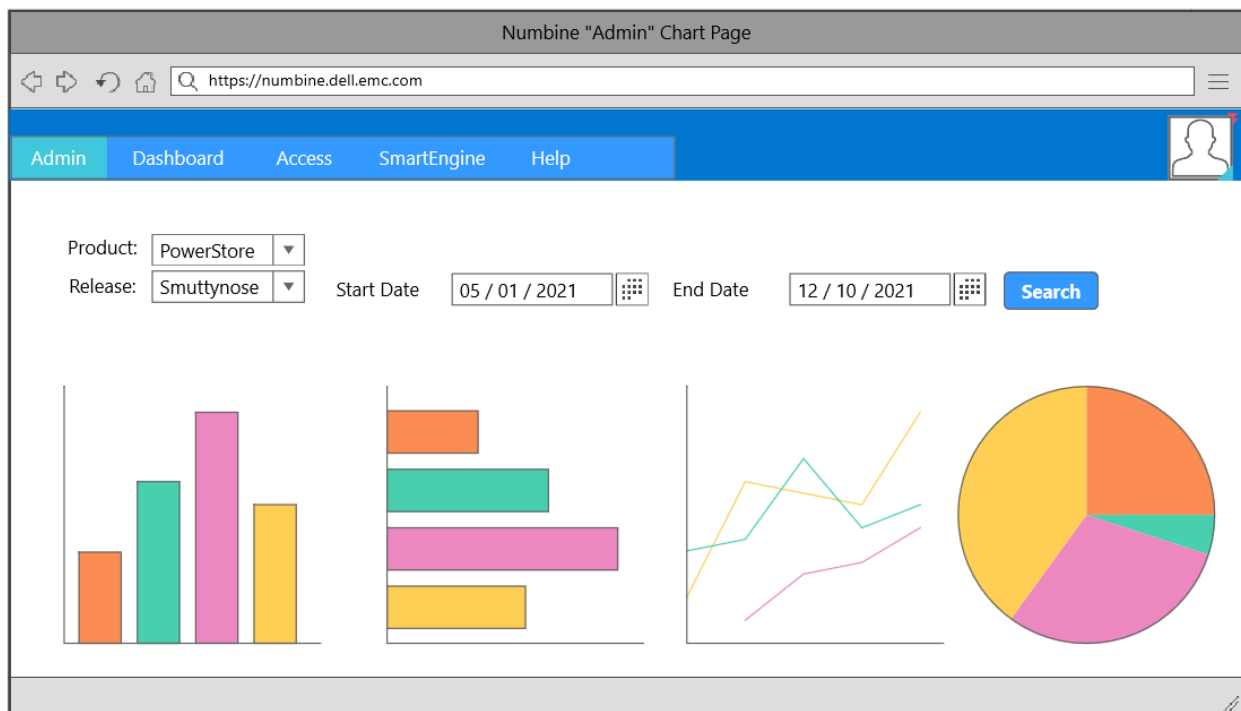
1) Admin Menu

Once a user is logged in, depending on their access privilege, it should see a menu bar relevant to her role. If it is admin role user logins, then it should display Admin.

Admin page contains Admin, Dashboard, Access and help menu options (SmartEngine is for future use).

2) Admin home page

This is the default page that is displayed to the Admin user after login.

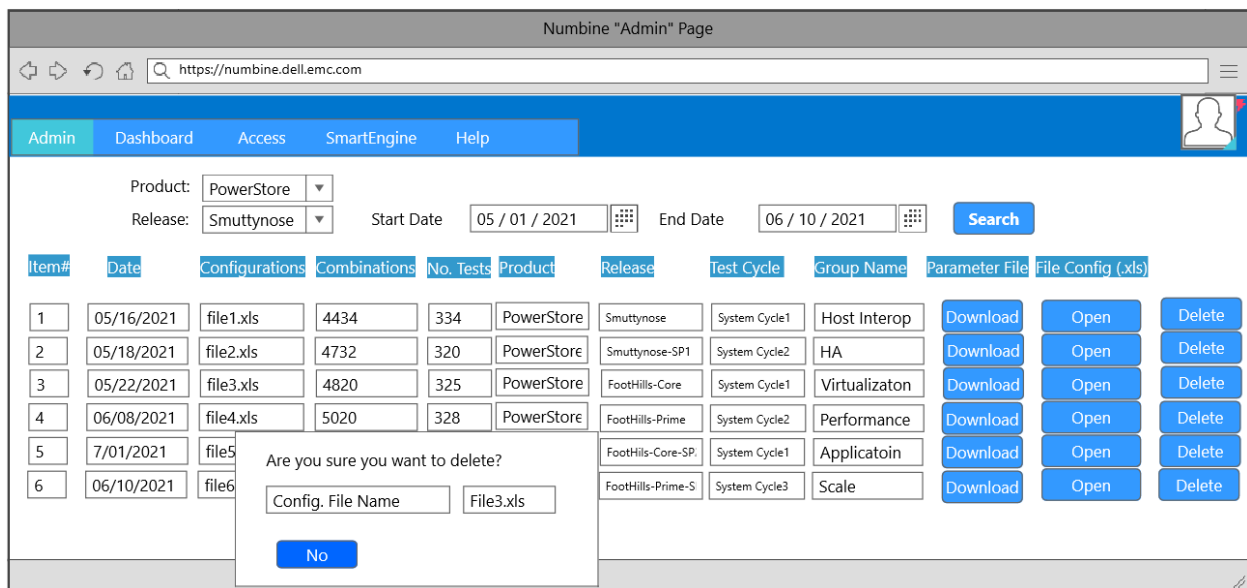


1. As an Admin I would like to be able to generate different types of charts using past data, current data, and future data for analysis.
2. As an admin I would like to filter the charts by using the following selections:
 - a. Select a product, or ALL PRODUCTS
 - b. Select a release relevant to the selected product or ALL
 - c. Select a date range, default start date 3 months ago and end date today
3. After the user presses the "Show charts" button (here it is called "search") the following charts are displayed:

- a. Activity report: As an admin I would like to monitor the users activity by their files creation
 - i. X date range grouped by weeks
 - ii. Y number of test files created in that week
- b. Users report: As an admin I would like to monitor the users using the system
 - i. X date range grouped by weeks
 - ii. Y number of users of that product(s) that have created a test file in that week
- c. Roles report
 - i. Pie chart divided by user roles, how many users have each role, this is not relevant to date range

3) Admin Dashboard Page

In “Dashboard” menu option, Admin user should be able to View all files generated by different users from different groups.



3.1 As an Admin I would like to be able to View all files that were generated by any user:

1. All those saved .xls files with the following data: number, date, filename, number of combinations, number of tests, product release, cycle, and group name (Note: cycle number and Group name are provided during user saved the files to database).
2. I should be able to filter the files by their date by using start and end date
3. I should be able to filter the files by Product, by default All products
4. I should be able to filter the files by Release. This option will be available only after selecting a product, by default it will be “All”

3.2 As an Admin user I would like to be able to Download the file

3.3 As an Admin user I would like to be able to Delete the file

1. Pressing the “Delete” button in the specific file row
2. The system will display a confirmation message
3. The system will delete the file

3.2 As an Admin user I would like to be able to Open the file - low priority

3.3 As an Admin user I would like to be able to sort the list by

1. date
2. product name
3. test cycle
4. group name

3) Admin Access Page

In “Access” menu option, Admin user should be able to CRUD users.

The screenshot shows a web browser window titled "A Web Page" with a URL bar containing "https://". The page has a navigation menu with links: [Admin](#), [Dashboard](#), [**Access**](#) (active), [SmartEngine](#), and [Help](#). The main content area is titled "Access" and contains a user management interface. At the top of this interface are controls: a dropdown menu for "All Roles" (with options "User", "Admin", and "Configurator"), a "Show active" toggle switch, a "Show In active" toggle switch, and a "New User" button. Below these controls is a table with the following data:

Name	Username(email)	Role	ProductGroups	Actions
Muhamad Ali	n.ali@dell.com	User	PowerStore PowerFlex	
Nivin Zaher	n.zaher@dell.com	Admin	PowerStore	
Shula Cohen	s.cohen@dell.com	Admin	PowerFlex PowerMax	
Yossi Cohen	y.cohen@dell.com	User	PowerFlex PowerMax	
Rani Jabarin	r.jabarin@dell.com	Configurator	PowerFlex PowerMax	

At the bottom of the table is a pagination control showing "« 1 2 3 4 »", with "1" highlighted.

3.1 As an Admin I would like to view all users in the system

1. I would like to be able to sort the list by Email, Role
2. I would like to be able to filter by Role
3. I would like to be able to view those that I have deleted (marked as not active)
4. by default, the users displayed will be ordered by their email and only active users are shown

3.2 As an Admin I would like to be able to Delete users by:

1. Pressing the “Delete” button in the specific user row
2. The system will display a confirmation message
3. The system will mark the user as “Not active”

3.3 As an Admin I would like to be able to Update users by:

1. Pressing the “Edit” button in the specific user row and the Udate user form will be displayed

3.4 As an Admin I would to be able to Add new users by pressing the “Add” button and the “Add User” form will be displayed

The screenshot shows a web browser window titled "A Web Page". The address bar contains "https://". The page has a navigation bar with links: [Admin](#), [Dashboard](#), [**Access**](#), [SmartEngine](#), and [Help](#). The main heading is "Access - New user/update user". Below this, there is a "Personal info" section with four text input fields: "Name", "Email", "Phone:", and "Password:". The "Email" field has a note below it: "This will be used as username". The "Password:" field has a note below it: "Must be 6 at least 6 letters". There is an "Active" toggle switch, currently in the "Off" position, with the text "In active" next to it. Below the toggle is a "Product groups" section with a dropdown menu. The dropdown menu is open, showing three options: "PowerStore", "PowerFlex", and "PowerMax". At the bottom of the form are two buttons: "Save" and "Cancel".

Admin Access Page - New/Update User

3.5 As an Admin user I would like to be able to Add and Update users while:

2. Assign their role : “admin” or “user” or “configurator”
3. Assign their “Product Group”, this can be by selecting from a list or by typing a new unique product group name that will be added once the user is saved
4. The “email” will be used as user name (not as shown below)
5. Pressing “New user” will open up the “New user” form

6. The system will mark the new user as Active
7. Raj: please define what happens if a user is updated to a different "Product Group"

3) User

As a user I would like to CRUD testing config generated files.

As a user, once I login I would like to see the "Dashboard" displayed by default.

As a user, I would like to have the following menu items: Dashboard, ConfigGenerate, Help.

1) User Menu

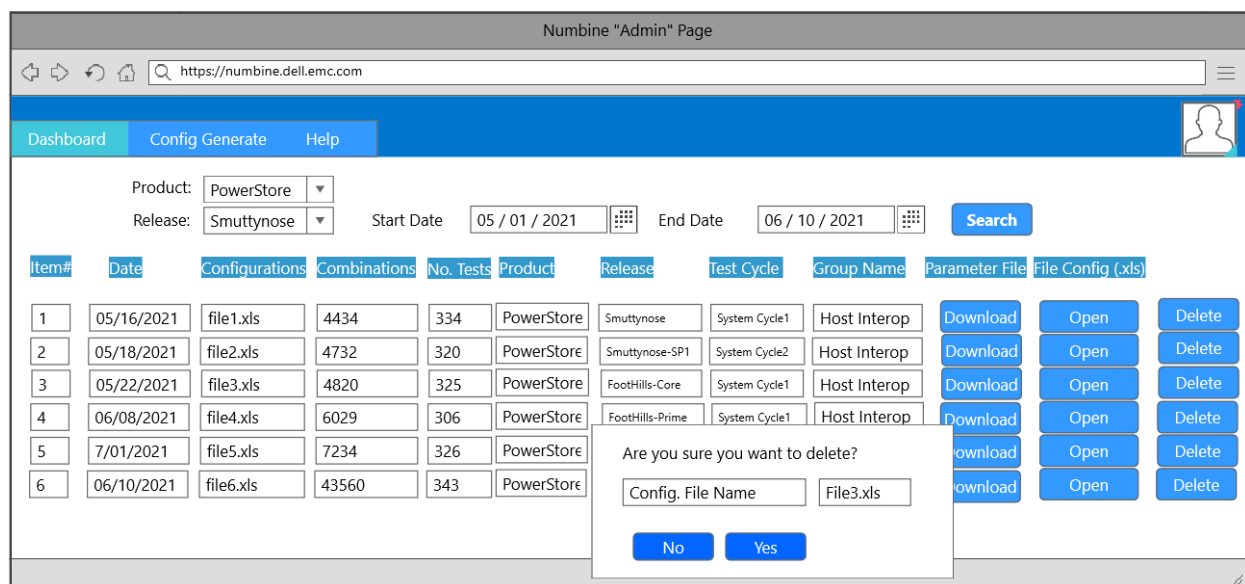
Once a user is logged in, depending on their access privilege, it should see a menu bar relevant to her role. If it is USER role, then it should display User dashboard and the User menu.

As a user I would like to have quick access to :

1. Dashboard menu: the list of files I have created (User dashboard and Home page and default page after login)
2. Config Generate menu: be able to create a new test config file (Config Generate)

2) User Dashboard page

The goal of this page is to allow the user to View, Delete and Download test configuration files that he have created.



1.1 As a user I would like to:

1. See a list of all Files that I have created and stored in the System.

2. For each file I would like to see the following fields: number, date, filename, number of combinations, number of tests, release, cycle, and group name (Note: Release, cycle number and Group name are provided during user saved the files to database)
3. To be able to filter the files by their date by using start and end date
4. To be able to filter the files by Product, by default All products will be displayed
5. To be able to filter files by Release. This option will be available only after selecting a product, by default it will be "All"

1.2 As a user I would like to be able to Download the file

7.3 As a user I would like to be able to Delete the file

1. Pressing the "Delete" button in the specific file row
2. The system will display a confirmation message
3. The system will delete the file

7.4 As user would like to be able to Open the file - low priority (you can download the file and open it)

3) User Config Generator page

The goal of this form is to allow the user to create new test configuration files.

The form is a complex form that acts like a wizard and follows a flow.

As a user I would like to create new test file by:

1. using the config params as they appear in the DB
2. using a file that includes these config params in a specific format (low priority)
3. write myself the content of the config params into the form (low priority)

The following will address only the first option.

As a user I would like to:

1. Step 1: Select to what I will be creating config file:
 - a. Select the product for which to create the config file. only one product can be selected
 - b. Select the release (and version displayed as one string)
2. Step 2: Fetch Only after selecting the above the “Fetch” button is enabled
 - a. Press the “Fetch” button and the configuration values of the selected product and version will be displayed
3. Step 3: Generate a new testing config file
 - a. Select the Tuple, default value is 2, available values are: 2, 3, 4
 - b. Only after Pressing “Fetch” button, the “Generate” button is enabled
 - c. Press the “Generate” button
 - d. System shall generate file name along with full path and Tuple number are passed as arguments to Microsoft PICT CLI tool
 - e. After pressing it, the config will be displayed
4. Step 4: Save the generated test config file
 - a. The user will select test cycle or enter a new value
 - b. Only after the “Generate” button us pressed, the “Save” button is enabled
 - c. Press the “Save” button
 - d. System shall save the config test file generated spreadsheet to database table as a file with the following info:
 - i. user id
 - ii. Product name
 - iii. Release
 - iv. Test cycle
 - v. Group name
 - vi. Used seed: 33868
 - vii. **Combinations: 24533**
 - viii. **Generated tests: 307**
 - ix. Generation time: 0:00:00
 - x. config parameters as a file/text
 - e. In case of success, a success message will appear to the user
5. alternate: Download
 - a. This is an optional step.
 - b. the user can download the fetched config values as a txt file
 - c. the user can download the test config generated values as xls file

4) Configurator

As a Configurator I would like to CRUD the configuration related to the products and their versions under my responsibility.

As a Configurator I would like to CRUD the users permissions to Products.

1) Configurator Menu

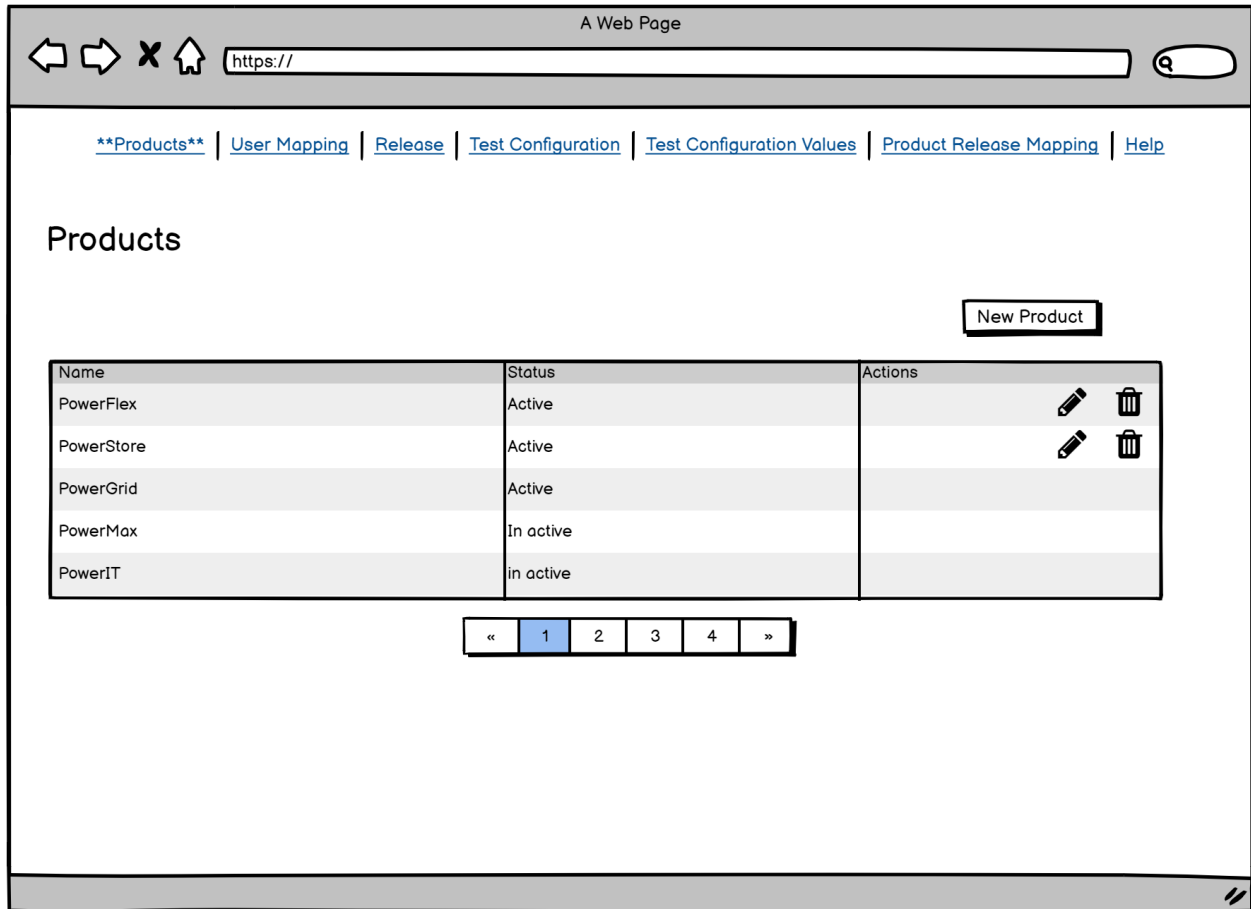
Once a user is logged in, depending on their access privilege, it should see a menu bar relevant to her role. If it is CONFIGURATOR role, then it should display Configurator dashboard and menu.

As a Configurator I would like to have quick access to :

1. CRUD products
2. CRUD versions
3. CRUD Configuration parameters
4. CRUD Configuration parameters values
5. CRUD product version configuration
6. CRUD users permissions to Products

2) Configurator Products page

The goal of this page is to allow the configurator to CRUD products.



2.1 As Configurator I would like to view all Products in the system with the following columns:

1. Product name, Product status
2. I would like to be able to sort the list by Product name, Product status
3. Once the Configurator enters this page the list is displayed, sorted by Product name.
4. The system will not display in any form other than this form inactive products

2.2 As Configurator I would like to be able to Add new Product:

1. Unique name
2. Default a new product is in “Active” status

2.2 As Configurator I would like to be able to Update a Product by pressing the button in the product row, the Update Product form will open:

1. The same form as “New Product”, with the following changes:
 - a. the table of releases will appear
 - b. if the Product is in active mode than the “Add Release” button is not available

- c. if the user changes the product status (in active to active or vise versa) than a confirmation message will appear

A Web Page

https://

[**Products**](#) | [User Mapping](#) | [Release](#) | [Test Configuration](#) | [Product Release Mapping](#) | [Help](#)

Products - New Product

Product name:

Active ☐ In active

Release	Version	Status
Smottynose	1.0	inactive
FH-Core	2.0	active
FH-Core-X	3.0	active

In case of Update Product, the table with the Product's releases will appear

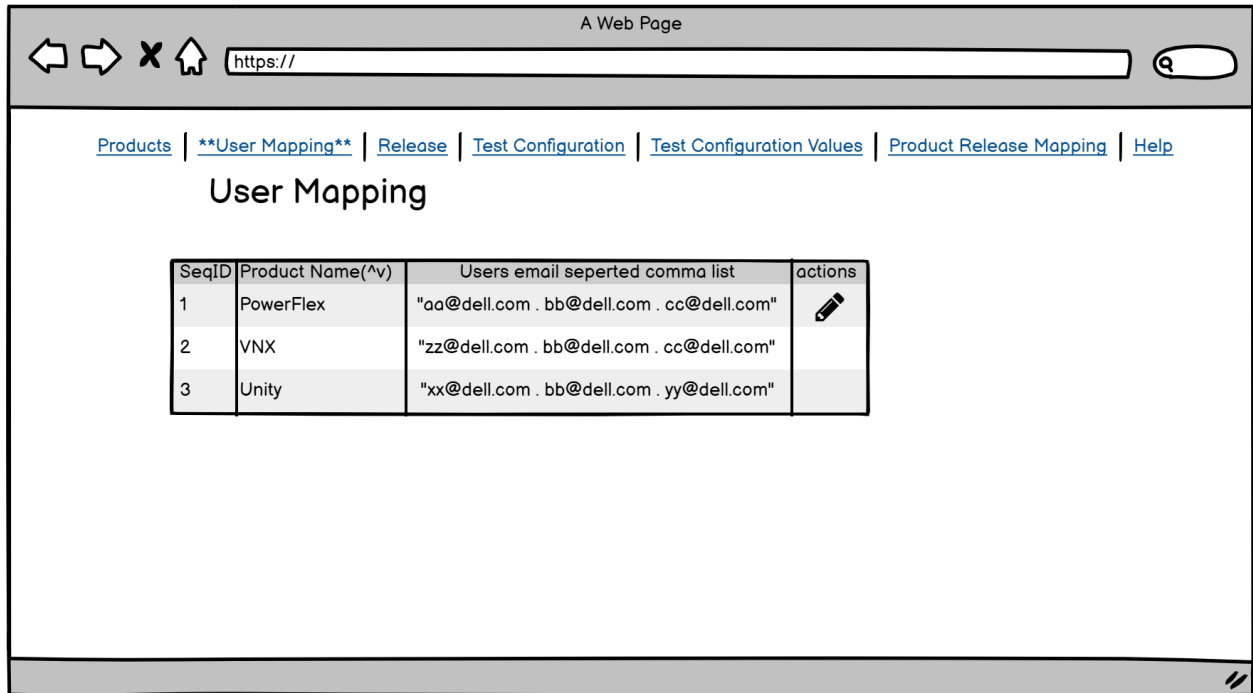
2.3 As Configurator I would like to be able to Delete a product. This is done by making the Product in status "In active"

1. The system will display a confirmation message once the user attempts to change the Product's status

3) Configurator User Mapping page

The goal of this page is to allow the configurator to CRUD relations between Users and Products.

The users will be able to view and access only Products that are mapped to them.



3.1 As Configurator I would like to view all Products and the users that are mapped to them with the following columns:

1. Product ID, Product name, list of Users
2. I would like to be able to sort the list by Product name, Product
3. Once the Configurator enters this page the list is displayed, sorted by Product name.

3.2 As Configurator I would like to be able to Update the user list related to a Product:

A Web Page


https://

[Products](#) | [**User Mapping**](#) | [Release](#) | [Test Configuration](#) | [Test Configuration Values](#) | [Product Release Mapping](#) | [Help](#)

User Mapping - Update User List

Product name: xxxxx

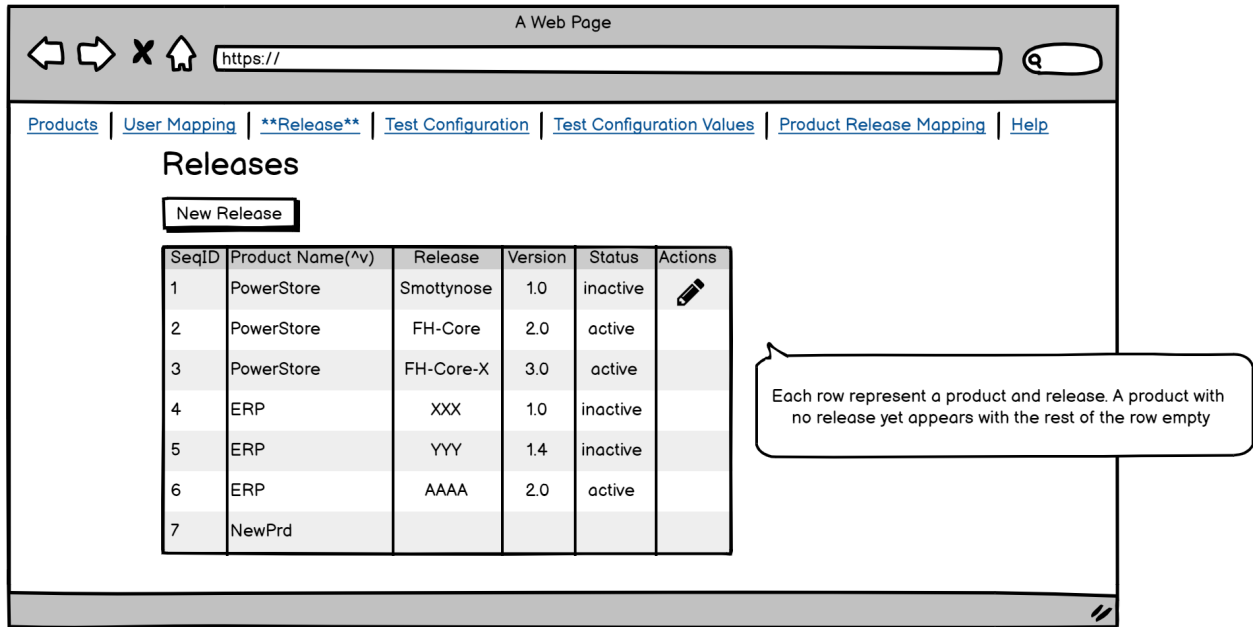
User List

SeqID	user email(^v)	actions
1	aa@dell.com	
2	zz@dell.com	
3	xx@dell.com	

1. Press the "Update" button of the Product from the "User Mapping" form
2. The system will display the "User Mapping - Update User List" form
3. The form will display for the specific product the list of users with their emails
4. As a Configurator I would like to be able to Add another User to the Product
 - a. Select a User from the list
 - b. Press the "Add" button
 - c. The form will display the Added user in the first row
5. As a Configurator I would like to be able to Remove a User from the Product
 - a. Press the Garbage button
 - b. The system will display "Are you sure?"
6. As a Configurator I would like to "Save" the changes I have done
 - a. Press the Save button
 - b. System will attempt to save the changes in DB
 - c. Success/Failure message will be displayed
7. As a Configurator I would like to "Quit" the changes I have done
 - a. System will display a "Are you sure?"

4) Configurator Releases Page

The goal of this page is to allow the configurator to CRUD products Releases.



4.1 As Configurator I would like to view all products Releases with the following columns:

1. Product name, Release, Version, Status

4.2 As Configurator I would like to update product Releases:

1. Press the “Update” button of the Product Release from the “Releases” form
2. The system will display the “Update Release” form

A Web Page

Products | User Mapping | ****Release**** | Test Configuration | Test Configuration Values | Product Release Mapping | Help

Release - Update Release

Product name: xxxxx

Release name:

Version:

Active ☐ In active

Release	Version	Status
Smottynose	1.0	inactive
FH-Core	2.0	active
FH-Core-X	3.0	active

Version is a composition of numbers with "." as a separator

3. The form will display the properties of the Product Release

4. The Configurator will be able to update any of the properties and Save OR Quit

4.3 As Configurator I would like to create and add new product Releases:

1. Press the "New Release" button from the "Releases" form
2. The system will display the "New Release" form

A Web Page

https://

[Products](#) | [User Mapping](#) | **[**Release**](#)** | [Test Configuration](#) | [Test Configuration Values](#) | [Product Release Mapping](#) | [Help](#)

Release - New Release

Product name:

Release name:

Version:

Active ☐ In active

Version is a composition of numbers with "." as a separator

Release	Version	Status
Smottynose	1.0	inactive
FH-Core	2.0	active
FH-Core-X	3.0	active

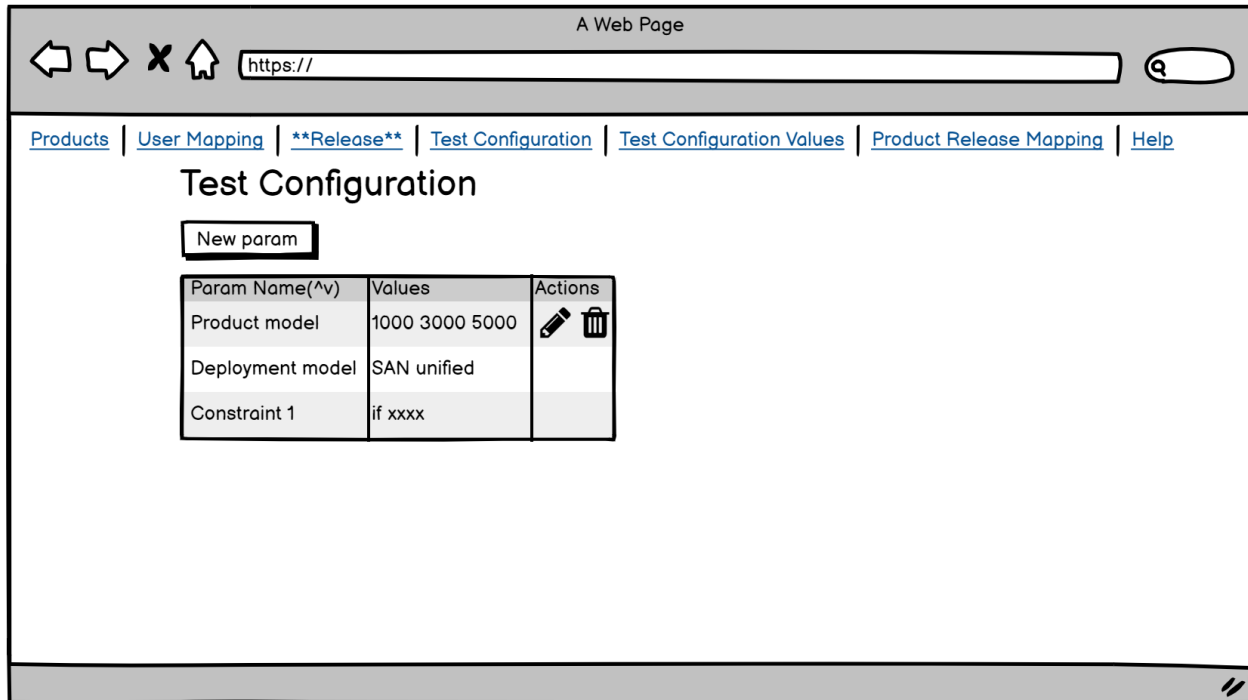
Once a product is selected, the table of existing releases is displayed

3. The Configurator will Select the Product and enter all values and Save OR Quit

4. Releases that are in active will be displayed only in this form and in the Releases list

5) Configurator Test Configuration Page

The goal of this page is to allow the configurator to CRUD Parameters and CRUD their values.



5.1 As Configurator I would like to view all parameters with the following columns:

1. Param name, list of its values
2. As a Configurator I would like to have the option to delete
 - a. A confirmation message will be displayed
 - b. note: Any existing files will continue to include the parameter

5.2 As a Configurator I would like to have the option to edit/new parameter

A Web Page

https://



[Products](#) | [User Mapping](#) | [Release](#) | **[**Test Configuration**](#)** | [Product Release Mapping](#) | [Help](#)

Test Configuration - New param

Param name:

Param description:

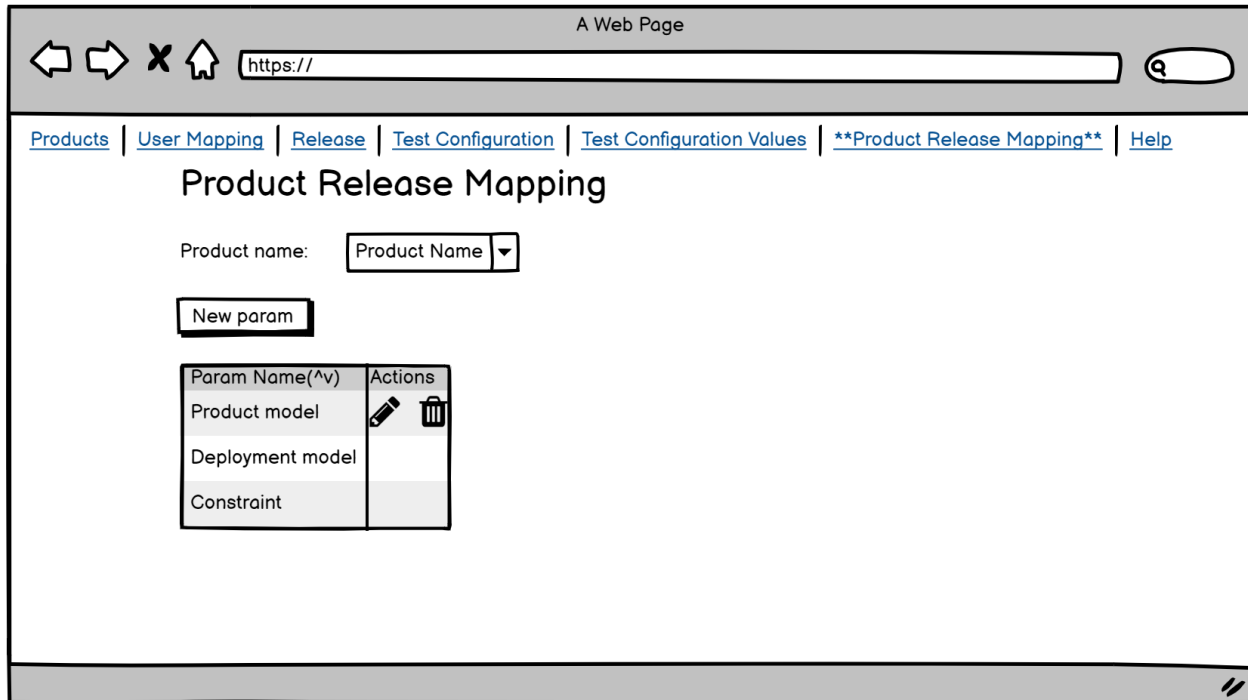
Param value:

SeqID	Value	actions
1	1000	 
2	3000	
3	5000	

1. The Update/New param form will open
2. Allow the Configurator to input param name and param description
3. Allow the Configurator to create a list of values
4. A param should include at least one value

6) Configurator Product Release Mapping Page

The goal of this page is to allow the configurator to Map parameters and their specific values to a specific Product release.



6.1 As Configurator I would like to view all parameters mapped to a specific Product:

1. I would like to select a Product
2. The system will display a list of all parameter names that are mapped to at least one of the Product releases

6.2 As Configurator I would like to add a new Param mapping to Product:

A Web Page

Products | User Mapping | Release | Test Configuration | Test Configuration Values | ****Product Release Mapping**** | Help

Product Release Mapping - New Param Mapping

Param name:

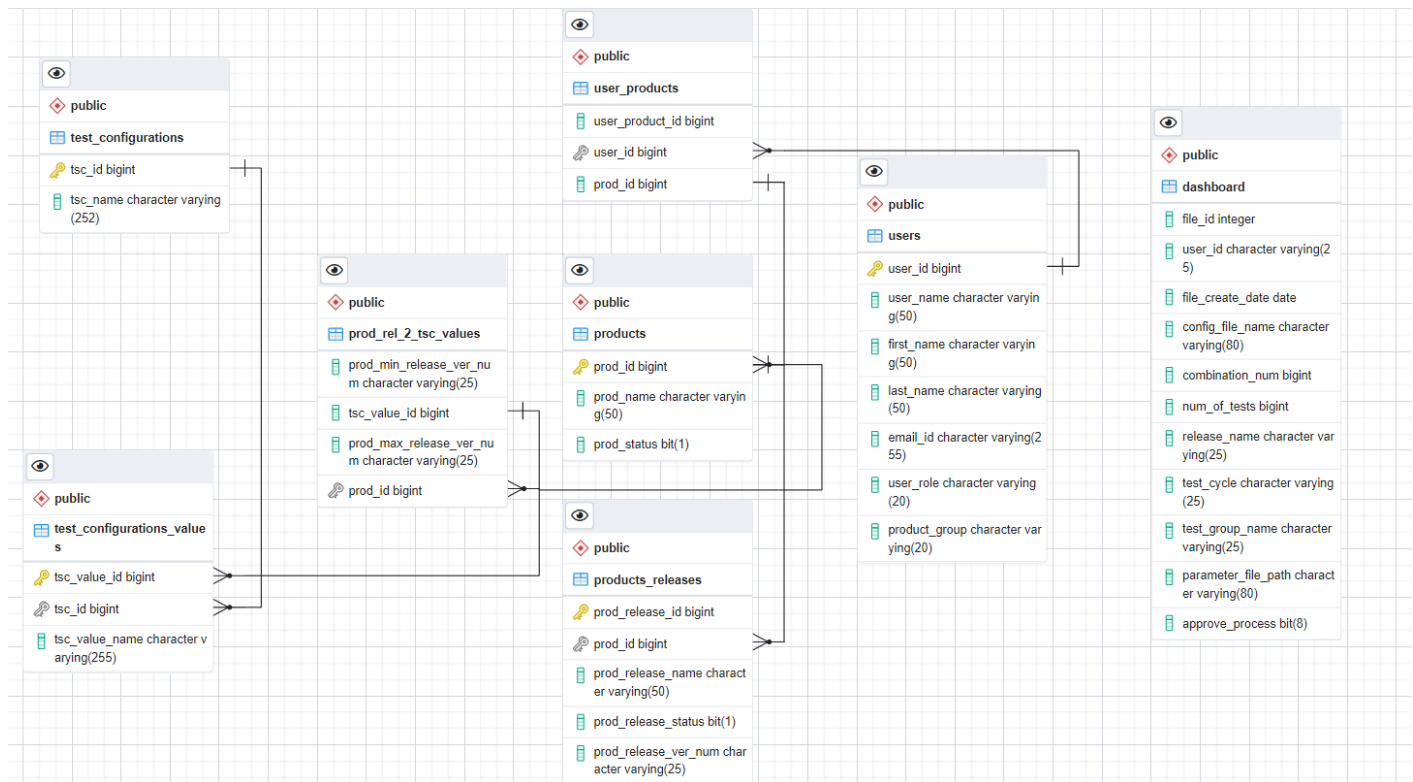
Once Param name is selected, the table is filled.
For update, the Param name is selected accordingly and the table is filled

Value	Start Release	End Release
1000	N/A	N/A
3000	1.0	ForEver
	1.1	1.0
5000	2.0	1.1
	2.2	2.0
		2.2

1. I would press the "New param" button and the following form will be displayed
2. It will allow me to select a parm to add
3. After selecting it the system will display all available values (also ones that are not yet mapped to any release of the product)
4. The system will allow me to select starting and ending release for each value of the parameter
5. The end release should be the same as or bigger that the starting release
6. The ending release can be "For ever"
7. The starting and ending releases can be "N/A" (not applicable) if the specific value is not relevant to any of the releases of the product

Technical notes

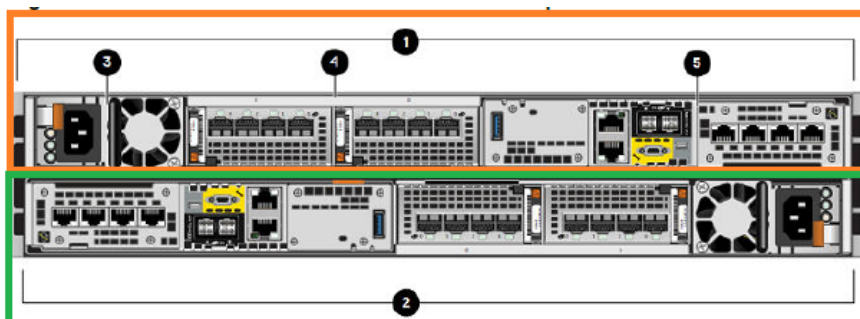
Numbine Application ER Diagram



Sample: What is DELLEMC PowerStore?

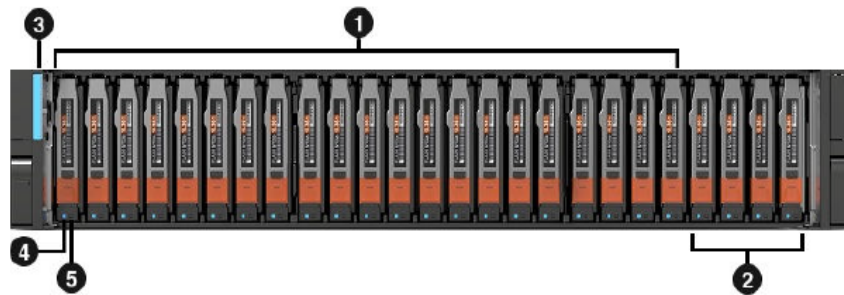
DELL EMC PowerStore is one of the latest midrange storage products that service low and medium size customers. It is scaleup and scale out system with the two node highly available appliance that can be configured as a cluster with the maximum of four appliances(total eight nodes). There are number of hardware and software components involves to delivers both block and file services.

Back View



- 1 – Node A (CoreOS/ESXi Server Installed)
- 2 – Node B (CoreOS/ESXi Server Installed)
- 3 – Power Supply Module
- 4 – I/O Module
- 5 – Embedded Module

Front View



- 1 – SSD or SCM NVMe drives
- 2 – NVRAM NVMe drives
- 3 – Base enclosure power on LED
- 4 – Drive ready/activity LED
- 5 – Drive fault LED

Hardware components:

Node is a service processor has dedicated number of sockets, memory, internal disks installed with a Linux based operating system to run proprietary software.

Each node consists of 4 port mezzanine card, management network port, service port, power adapter, fan and two SLICS

SLIC: Small I/O Card – contained within SP or DM. SLIC options include 10GbE, 25GBE, 16Gig FC, 32Gig FC, Fibre Channel, iSCSI, SAS. SAS: Serial Attached SCSI – A point-to-point serial protocol for moving data to drives. A higher speed serial version of SCSI.

SP: Storage Processor – provides access to all external hosts and the file side of PowerStore.

Node A (SP A: Storage processor A) – Generic term for the first storage processor in PowerStore for block.

Node B(SP B: Storage processor B) – Generic term for the second storage processor in PowerStore for block.

SPE: Storage Processor Enclosure.

DAE: Disk Array Enclosure.

DPE: Disk and Processor Enclosure, containing both storage processors and disk drives.

Tables content

Table Name: users

```
numbine=# \d+ users
```

Column	Type	Collation	Nullable	Table "public.users" Default	Storage	Stats target	Description
user_id	bigint		not null	nextval('users_user_id_seq'::regclass)	plain		
user_name	character varying(50)		not null		extended		
first_name	character varying(50)				extended		
last_name	character varying(50)				extended		
email_id	character varying(255)				extended		
user_role	character varying(20)				extended		
product_group	character varying(20)				extended		

Indexes:
"users_pkey" PRIMARY KEY, btree (user_id)
Access method: heap

```
numbine=# select * from users;
```

	user_id	user_name	first_name	last_name	email_id	user_role	product_group
1	smithj	Smith	John	smith.john@dell.com	user	PowerStore	
2	Williamsb	Williams	Brown	Williams.brown@dell.com	admin	PowerStore	
3	Willonj	Wilson	Jones	Wilson.jones@dell.com	admin	PowerMax	
4	Taylorom	Taylor	Moore	Taylor.moore@dell.com	user	PowerFlex	
5	Thomasa	Thomas	Anderson	Thomas.Anderson@dell.com	admin	PowerFlex	

Table Name: Products

```
numbine=# \d+ products
```

Column	Type	Collation	Nullable	Table "public.products" Default	Storage	Stats target	Description
prod_id	bigint		not null		plain		
prod_name	character varying(50)		not null		extended		
prod_status	bit(1)				extended		

Indexes:
"products_pkey" PRIMARY KEY, btree (prod_id)
Referenced by:
TABLE "prod_rel_2_tsc_values" CONSTRAINT "prod_rel_2_tsc_values_prod_id_fkey" FOREIGN KEY (prod_id) REFERENCES products(prod_id) NOT VALID
Access method: heap

```
numbine=# select * from products order by prod_id;
```

	prod_id	prod_name	prod_status
10	CLARiiON	0	
20	VNX	0	
30	VNXe	0	
50	Unity	1	
100	PowerStore	1	
200	PowerMAX	1	
300	PowerScale	1	
400	PowerFlex	1	

Table Name: users_products

```
numbine=# \d+ user_products
Table "public.user_products"
  Column      | Type   | Collation | Nullable | Default | Storage  | Stats target | Description
-----+-----+-----+-----+-----+-----+-----+-----
 user_product_id | bigint |           |          |          | plain    |              |
 user_id        | bigint |           |          |          | plain    |              |
 prod_id        | bigint |           |          |          | plain    |              |
Access method: heap
```

```
numbine=# select * from user_products;
 user_product_id | user_id | prod_id
-----+-----+-----
                2 |        1 |    100
                3 |        1 |    200
                4 |        2 |    200
                5 |        2 |    300
```

Table Name: products_releases

```
numbine=# \d+ products_releases
Table "public.products_releases"
  Column          | Type          | Collation | Nullable | Default | Storage  | Stats target | Description
-----+-----+-----+-----+-----+-----+-----+-----
 prod_release_id | bigint        |           | not null |          | plain    |              |
 prod_id         | bigint        |           | not null |          | plain    |              |
 prod_release_name | character varying(50) |           | not null |          | extended |              |
 prod_release_status | bit(1)       |           |          |          | extended |              |
 prod_release_ver_num | character varying(25) |           |          |          | extended |              |
Indexes:
    "products_releases_pkey" PRIMARY KEY, btree (prod_release_id)
Access method: heap
```

```
numbine=# select * from products_releases order by prod_release_id;
 prod_release_id | prod_id | prod_release_name | prod_release_status | prod_release_ver_num
-----+-----+-----+-----+-----
                1 |    100 | smuttynose        | 1                   | v1.0
                2 |    100 | smuttynose-sp1    | 1                   | v1.1
                3 |    100 | smuttynose-sp2    | 1                   | v1.2
                4 |    100 | smuttynose-sp3    | 1                   | v1.3
                5 |    100 | smuttynose-sp4    | 1                   | v1.4
                6 |    100 | FH-Core           | 1                   | v2.0
                7 |    100 | FH-Core-SP1       | 1                   | v2.1
                8 |    100 | FH-Core-SP2       | 1                   | v2.2
                9 |    100 | FH-Core-SP3       | 1                   | v2.3
               10 |    100 | FH-Core-SP4       | 1                   | v2.4
               11 |    100 | FH-Prime          | 1                   | v3.0
               12 |    100 | FH-Prime-SP1      | 1                   | v3.1
               13 |    100 | FH-Prime-SP2      | 1                   | v3.2
               14 |    100 | FH-Prime-SP3      | 1                   | v3.3
               15 |    100 | FH-Prime-SP4      | 1                   | v3.4
               16 |    100 | Victory           | 1                   | v4.0
               17 |    100 | Victory-SP1       | 1                   | v4.1
               18 |    100 | Victory-SP2       | 1                   | v4.2
               19 |    100 | Victory-SP3       | 1                   | v4.3
               20 |    100 | Victory-SP4       | 1                   | v4.4
```

Table Name: prod_rel_2_tsc_values (Product Release Test Configuration Values)

```
numbline=# \d prod_rel_2_tsc_values
Table "public.prod_rel_2_tsc_values"
  Column          |          Type          | Collation | Nullable | Default
-----+-----+-----+-----+-----
 prod_min_release_ver_num | character varying(25) |           | not null |
 tsc_value_id       | bigint                |           | not null |
 prod_max_release_ver_num | character varying(25) |           |          |
 prod_id            | bigint                |           |          |
Foreign-key constraints:
 "prod_rel_2_tsc_values_prod_id_fkey" FOREIGN KEY (prod_id) REFERENCES products(prod_id) NOT VALID
```

```
numbline=# select * from prod_rel_2_tsc_values;
 prod_min_release_ver_num | tsc_value_id | prod_max_release_ver_num | prod_id
-----+-----+-----+-----
 1.0                      | 1            | 3.0                      | 100
 1.0                      | 2            | 3.0                      | 100
 1.0                      | 3            | 3.0                      | 100
 1.0                      | 4            | 3.0                      | 100
 1.0                      | 5            | 3.0                      | 100
 1.0                      | 6            | 3.0                      | 100
 1.0                      | 7            | 3.0                      | 100
 1.0                      | 8            | 3.0                      | 100
 1.0                      | 9            | 3.0                      | 100
 1.0                      | 10           | 3.0                      | 100
```

Table Name: test_configurations

```
numbline=# \d+ test_configurations
Table "public.test_configurations"
  Column |          Type          | Collation | Nullable |          Default          | Storage | Stats target | Description
-----+-----+-----+-----+-----+-----+-----+-----
 tsc_id  | bigint                |           | not null | generated always as identity | plain   |              |
 tsc_name | character varying(252) |           | not null |                               | extended |              |
Indexes:
 "test_configurations_pkey" PRIMARY KEY, btree (tsc_id)
Referenced by:
 TABLE "test_configurations_values" CONSTRAINT "tsc_id" FOREIGN KEY (tsc_id) REFERENCES test_configurations(tsc_id) ON UPDATE CASCADE ON DELETE CASCADE NOT VALID
Access method: heap
```

```

numbine=# select * from test_configurations;
tsc_id |          tsc_name
-----+-----
 1 | Array Model
 2 | Deployment Type
 3 | Product Type
 4 | Number of Appliance in Cluster
 5 | MultiPath_Type
 6 | Host_OS
 7 | Browsers
 8 | Locale
 9 | Server Infra
10 | Slic0
11 | Slic1
12 | HBA Connectivity
13 | vCenter Server
14 | MezzCard0
15 | MezzCard1
16 | SDNAS
17 | Node ESXi Version
18 | Switch Status
19 | DPE Disk Type
20 | DPE Disk Size
21 | Local Disk Size
22 | NVRAM Numbers
23 | RAID Width
24 | Vol Migration
25 | Application Config
26 | Applications
27 | File System
28 | DAE Type
29 | NVRAM Type
30 | Test Priority
31 | Source Appliance HW Type
32 | Target Appliance HW Type
33 | Constraints

```

```

numbine=# select tsc_name, tsc_value_name from prod_rel_2_tsc_values pr inner join test_configurations_values tcv on pr.tsc_value_id=tcv.tsc_value_id inner join test_configurations tc on tcv.tsc_id=tc.tsc_id where prod_id=100 and prod_min_release_ver_num='1.0';
tsc_name | tsc_value_name
-----+-----
Array Model | 1000
Array Model | 3000
Array Model | 5000
Array Model | 7000
Array Model | 9000
Deployment Type | SAN
Deployment Type | SAN Only
Deployment Type | Unified(10)
Deployment Type | NAS Only
Deployment Type | AppsOn(9)
Product Type | PowerStoreT
Product Type | PowerStoreX
Product Type | Riptide
Number of Appliance in Cluster | 1
Number of Appliance in Cluster | 2
Number of Appliance in Cluster | 3
Number of Appliance in Cluster | 4
MultiPath_Type | NMP
MultiPath_Type | PowerPath
MultiPath_Type | PPVE
MultiPath_Type | Infoscalle
MultiPath_Type | MPxIO
MultiPath_Type | MPIO
MultiPath_Type | DM-MPIO
Host_OS | ESXi7.0
Host_OS | ESXi6.7U3b
Host_OS | AIX7.2
Host_OS | CentOS
Host_OS | RHEL8.0
Host_OS | SLES12-SP4
Host_OS | SLES15
Host_OS | SLES15-SP1
Host_OS | Windows Server 2016
Host_OS | Windows Server 2019
Host_OS | Windows10
Host_OS | Ubuntu 18.04
Host_OS | Ubuntu 19.04

```