



TECNOLÓGICO DE MONTERREY
Querétaro Campus

Professors:

Ricardo Cortés Espinosa
Eduardo Daniel Juárez Pineda

Software Construction and Decision Making

Project Progress Report 4

Team Members:

Karla Alejandra Padilla González A0170331
Daniel Gutiérrez Gómez A01068056
Azul Rosales A01706348
José Pablo Martínez Valdivia A01275676
Alejandra Cabrera Ruiz A01704463
Salvador Rodríguez Paredes A01704562
Carlos Isaac Dávalos Lomelí A01706041

Delivery Date:

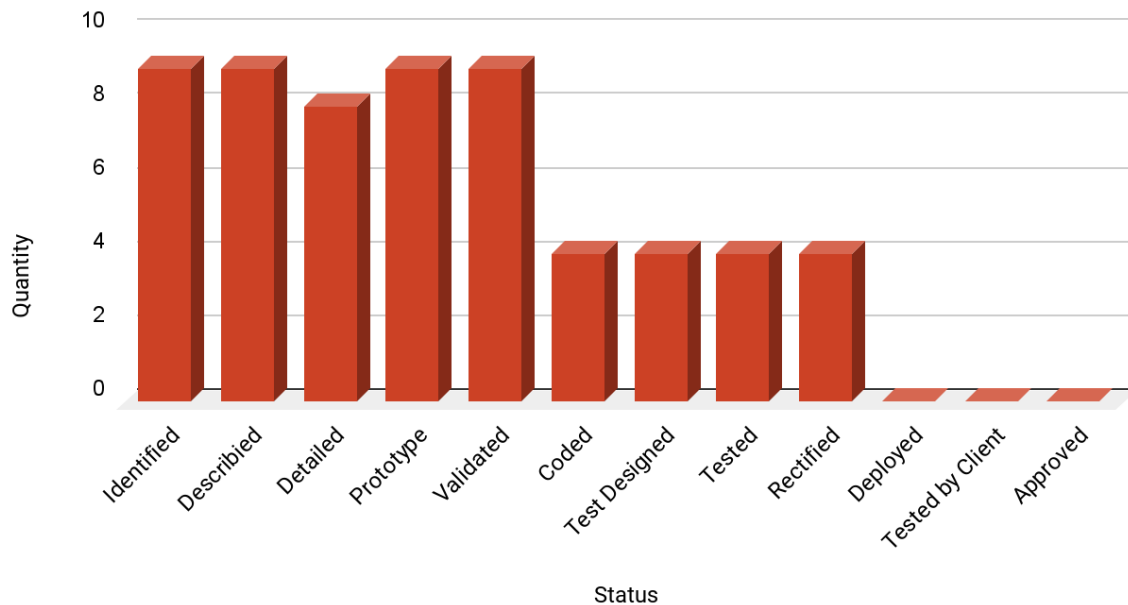
31/03/2023

A0170331@tec.mx
A01068056@tec.mx
A01706348@tec.mx
A01275676@tec.mx
A01704463@tec.mx
A01704562@tec.mx
A01706041@tec.mx

Functional Requirements Implementation

4 out of 9 functional requirements were implemented, making up for 44.4% of the project progress.

Use Cases Overview



The following functions (marked in red) were implemented:

Number	Requirement	Priority	Difficulty	Stability	Risk
1	Import CSV	4	4	2	4
2	Visualize dashboard	4	4	3	3
3	Authentication	4	3	4	4
4	Create member	3	2	4	1
5	Assign member to project	3	2	4	1
6	Modify member capacity	3	1	3	1
7	Create project	4	2	4	2
8	Log out	3	1	4	1
9	View list of projects	3	3	3	3

After completing the Authentication and Logout functions, we decided to implement the Create Project and Import CSV features, as they are critical to the development of the Project Progress Dashboard.

Tests Design and Execution

To ensure the reliability of the functionalities in our web app, we've designed and executed test cases for each one of them. This approach allows us to identify and rectify any issues, ensuring a seamless user experience.

Import CSV

1. **Use case description:**

ID: U.C. 01

Name: Import CSV

Actor(s): User

Description: This function allows the user to interact with our system by submitting a CSV file, a file which has all the information from which our system proceeds to analyze.

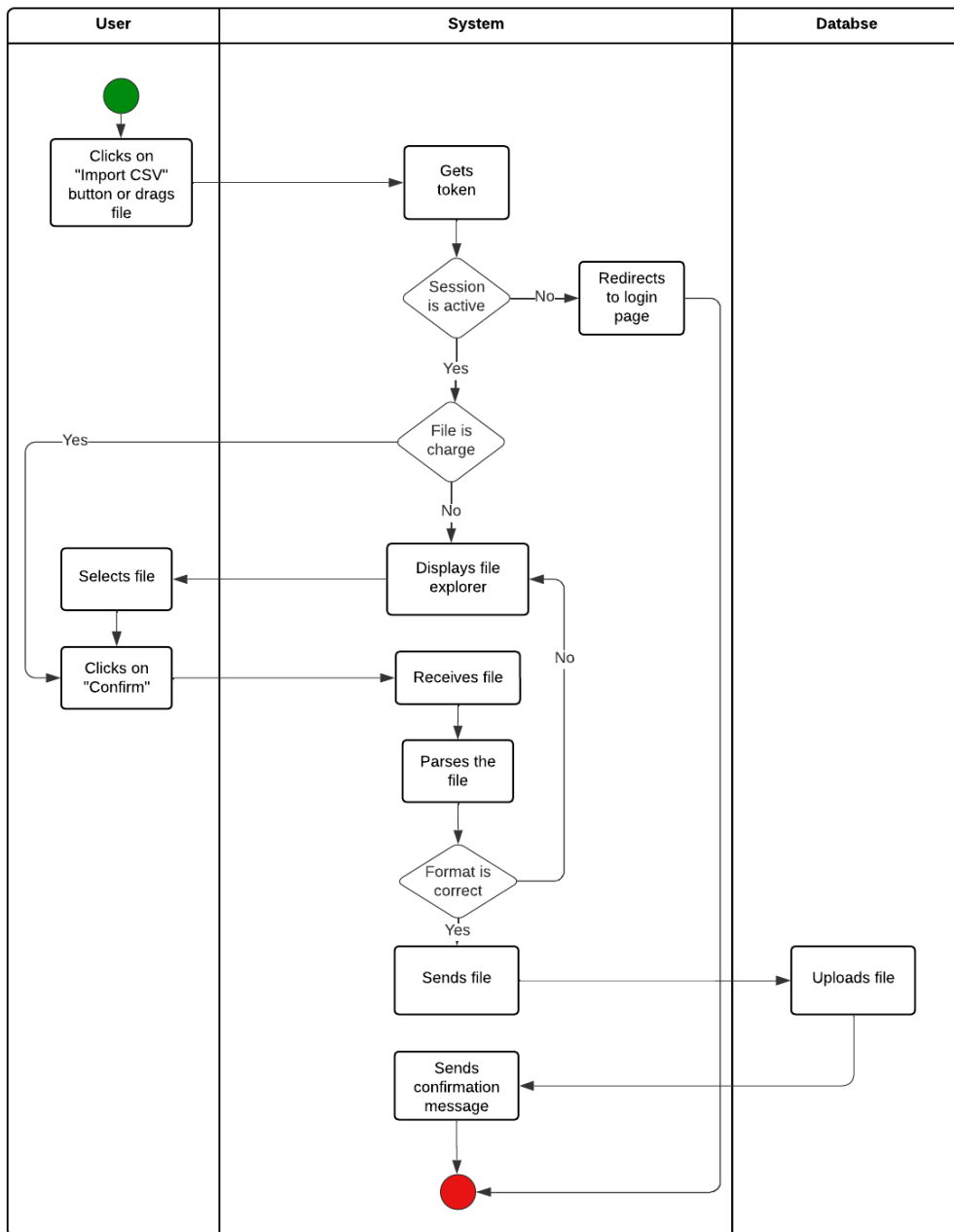
Pre-conditions: User having a login session.

Post-conditions: System shows confirmation message of uploaded file.

Non-functional requirements:

1. Security: User has to be authenticated by Auth0.

Import CSV



Alternate Flows

- A1→ If the session is not active, system redirects to login page
- A2→ System validates if file has been dragged, if not continue basic flow
- A3→ System validates if array lengths are the same
- A4→ System verifies if fields are the same with mock format

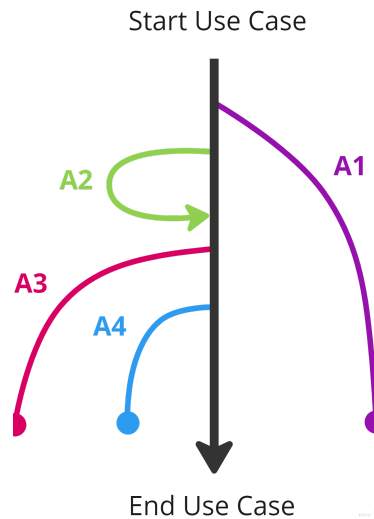


Table 1. Partial Scenario Matrix for the Import CSV Use Case

Scenario Name	Starting Flow	Alternate
Scenario 1 - Not active session	Basic Flow	A1
Scenario 2- File dragged	Basic Flow	A2
Scenario 3 - Not same array length	Basic Flow	A3
Scenario 4 - Not correct fields	Basic Flow	A4

Table 2. Test Case Matrix for Import CSV Use Case

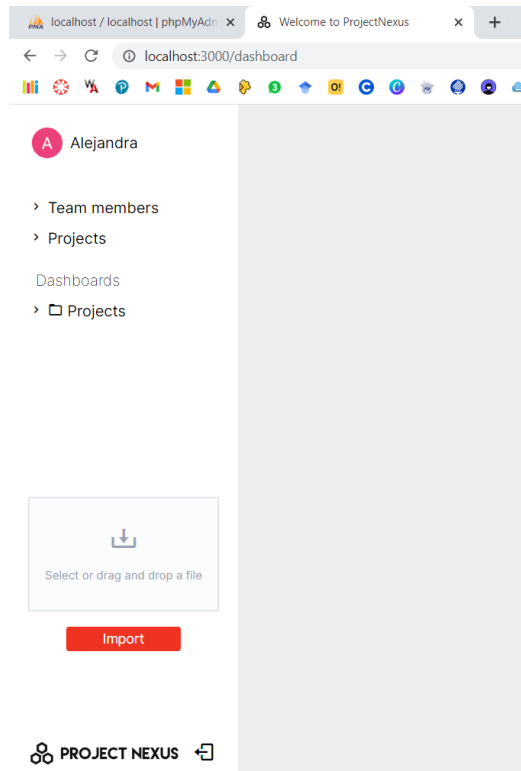
Test Case ID	Scenario / Condition	Active Token	File Dragged	Same array length	Same fields	Expected Result
TC 1.1	Scenario 1 - Not active session	F	N/A	N/A	N/A	Redirects to login page
TC 1.2	Scenario 2- File dragged	T	T	N/A	N/A	Jumps to 1.7
TC 1.3	Scenario 3 - Not same array length	T		F	N/A	Prompt error message and redirects to default page
TC 1.4	Scenario 4 - Not	T		T	F	Prompt error

	correct fields					message and redirects to default page
--	----------------	--	--	--	--	---------------------------------------

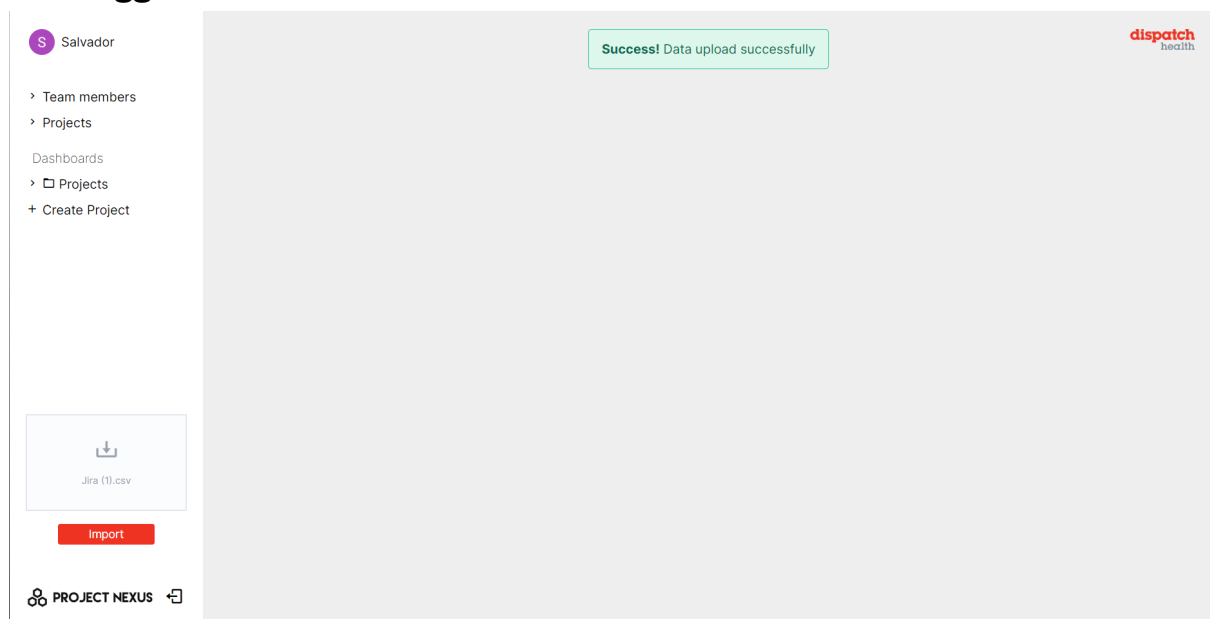
Table 3. Test Case Matrix with Data Values

Test Case ID	Scenario / Condition	Active Token	File Dragged	Same array length	Same fields	Expected Result
TC 1.1	Scenario 1 - Not active session	No	N/A	N/A	N/A	Redirects to login page
TC 1.2	Scenario 2- File dragged	Yes	Yes	N/A	N/A	Jumps to 1.7
TC 1.3	Scenario 3 - Not same array length	Yes		No	N/A	Prompt error message and redirects to default page
TC 1.4	Scenario 4 - Not correct fields	Yes		Yes	No	Prompt error message and redirects to default page

Interface:



File dragged:



Not same array length :

S

Salvador

> Team members

> Projects

Dashboards

>

Projects

+ Create Project

↓

notSameLenght.csv

Import

PROJECT NEXUS

Error! Rows aren't the same lenght

dispatch health

Not correct fields:

S

Salvador

> Team members

> Projects

Dashboards

>

Projects

+ Create Project

↓

invalidHeaders.csv

Import

PROJECT NEXUS

Error! Row headers don't match

dispatch health

Create Project

1. Use case description:

ID: U.C. 07

Name: Create project

Actor(s): User

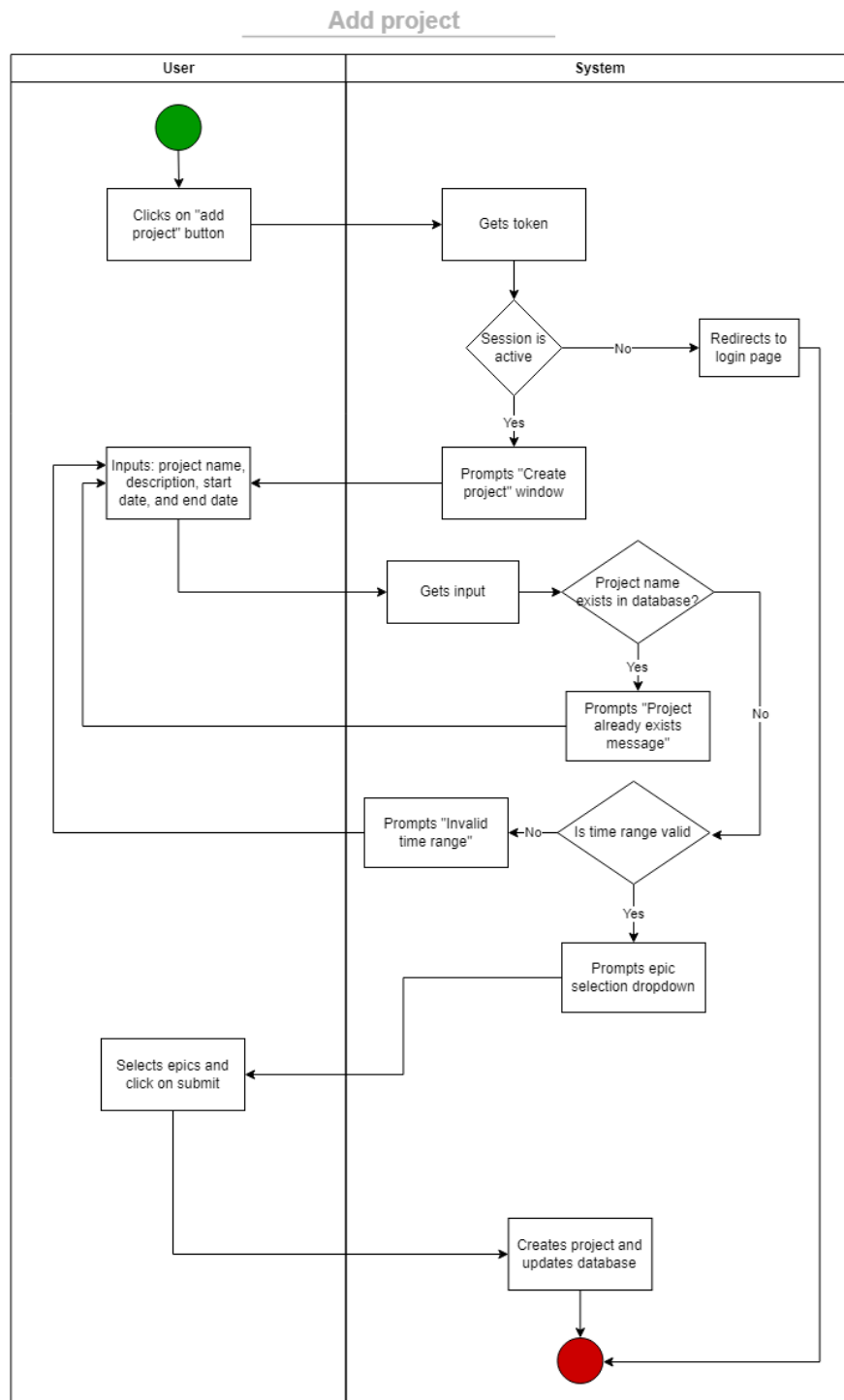
Description: User can create a new project and assign pre created epics

Pre-conditions: User has already import csv

Post-conditions:

Non-functional requirements:

1. Security: User has to be authenticated by Auth0



Events flow	
Normal Course	Alternatives
7.1 User clicks on create project button	
7.2 System gets user tokens	
7.3 System validates if session is active	7.3.1 If not, system redirects user to login page
7.4 System prompts "Create project" window	
7.5 User inputs: "project name", "description", "start date" and "end date"	
7.6 System gets user's input	
7.7 System validates if project is not already in database	7.7.1 If project already exists, system prompts alert, "project already exists" and return to 7.4, if not continue
7.8 System validates if project is inside a valid time range	7.8.1 If not, system prompts "invalid time range" and return to 7.4, else continue
7.9 System prompts epic's selection dropdown	
7.10 User selects epics	
7.11 User clicks on confirm	
7.12 System creates project and updates database	

Alternate Flows

A1→ If the session is not active, system redirects to login page

A2→ If project is already in database, system prompts error message

A3→ If project is not inside a valid time range, system prompts error message

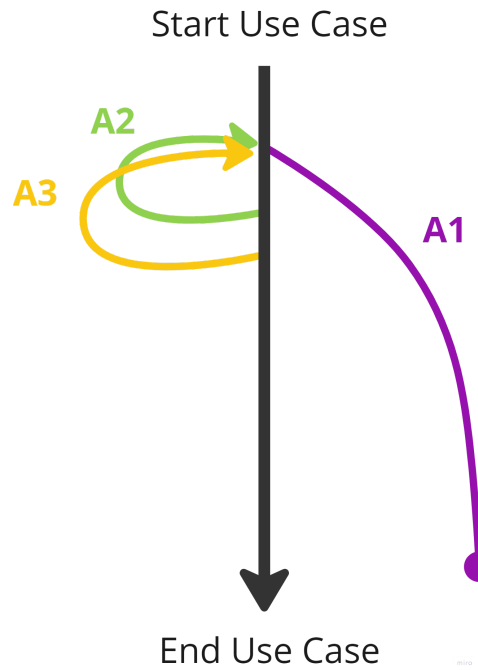


Table 1. Partial Scenario Matrix for the Create Project Use Case

Scenario Name	Starting Flow	Alternate
Scenario 1 - Session not active	Basic Flow	A1
Scenario 2 - Project exists	Basic Flow	A2
Scenario 3 - Invalid time range	Basic Flow	A3

Table 2. Test Case Matrix for Create Project Use Case

Test Case ID	Scenario / Condition	User authenticated	Active Token	Window for input	Project name in database	Project is in time valid range	Select epic	Expected Result
TC 7.1	Scenario 1 - Not active session	T	F	N/A	N/A	N/A	N/A	Redirects to login page
TC 7.2	Scenario 2- Project already exists	T	T	T	T	N/A	N/A	Prompt alert "project already exists" and return to 7.4
TC	Scenario	T	T	T	F	F	F	Prompt

7.3	3 - Invalid time range							error message and redirects to default page
TC 7.4	Scenario 4 -Successful creation of project	T	T	N/A	F	T	N/A	Prompt message of successful project creation

Table 3. Test Case Matrix with Data Values for Create Project Use Case

Test Case ID	Scenario / Condition	User authenticated	Active Token	Window for input	Project name, start_date, end_date,	Selects epics	Expected Result
TC 7.1	Scenario 1 - Not active session	bernardo@dispatc health.com	<random(32)> for 5 minutes	N/A	N/A	N/A	Redirects to login page
TC 7.2	Scenario 2- Project already exists	bernardo@dispatc health.com	<random(32)> for 5 minutes.	name, description, start_date, end_date	<'Advanced employee management system', '2023-03-13', '2023-05-01', ">	N/A	Prompt alert "project already exists" and return to 7.4
TC 7.3	Scenario 3 - Invalid time range	T	<random(32)> for 5 minutes	name, description, start_date, end_date	<'Advanced employee management system', '2023-03-13', '2023-05-01', ">	F	Prompt error message and redirects to default page
TC 7.4	Scenario 4 -Successful creation of project	T	<random(32)> for 5 minutes	N/A	<'Sentiment analysis for product rating', '2023-03-16', '2023-05-03', ">	<'Part-23 4,Part-23 5,'>	Prompt message of successful project creation

Clicks with valid session

The screenshot shows a web application interface for 'ProjectNexus' at 'localhost:3000/dashboard'. The user 'José' is logged in. The left sidebar contains navigation links: 'Team members', 'Projects', 'Dashboards', 'Projects' (with sub-links 'Express Tech Excellence' and 'Nuevo proyecto'), and a '+ Create Project' button. The main content area is empty. The right sidebar shows the 'Application' tab with a table of cookies:

Name	Value	D..	P..	E..	S..	H..	S..	P..
TP_JAR	2023-03-07-21	-	/	2..	19	✓	N..	M..
appSession	eyJhbGciOiJIUzI1LCJ...	L..	/	2..	2..	✓	L..	M..
connect.sid	s%3AapoQhCxt2Kx8...	L..	/	S..	91	✓		M..

The 'Console' tab shows two DevTools messages:

```
DevTools failed to load source map: Could not load content for chrome-extension://cfhdbjthkthekdabdcdddlifdgb/browser-polyfill.js.map: System error: net::ERR_FILE_NOT_FOUND
DevTools failed to load source map: Could not load content for http://localhost:3000/scripts/index.min.js.map: HTTP error: status code 404, net::ERR_HTTP_RESPONSE_CODE_FAILURE
```

The bottom status bar shows 'localhost:3000/createProject'.

The screenshot shows the 'Create Project' form in the 'ProjectNexus' application. The 'dispatch health' logo is in the top right. The form has fields for 'Name', 'Start date' (dd/mm/yyyy), and 'End date' (dd/mm/yyyy). Below these is a 'Select your Project epics' dropdown and a 'Save' button. The left sidebar is identical to the previous screenshot. The right sidebar shows the 'Application' tab with the same cookie table as above. The 'Console' tab shows the same DevTools messages as above. The bottom status bar shows 'localhost:3000/createProject'.

Clicks with invalid session

Welcome to ProjectNexus

localhost:3000/createProject

José

Team members

Projects

Dashboards

Projects

Express Tech Excellence

Nuevo proyecto

Create Project

Select or drag and drop a file

Import

dispatch

health

Create Project

Express Tech Excellence

Start date

01/03/2023

End date

31/03/2023

Select your Project epics

☒ PART-234

☐ PART-289

☒ PART-306

Save

Welcome to ProjectNexus

localhost:3000/createProject

José

Team members

Projects

Dashboards

Projects

Express Tech Excellence

Nuevo proyecto

Create Project

Select or drag and drop a file

Import

dispatch

health

Create Project

Project already exists

Name

Start date

dd/mm/yyyy

End date

dd/mm/yyyy

Select your Project epics

Save

Invalid time range

Welcome to ProjectNexus

localhost:3000/createProject

Actualizar

José

Team members

Projects

Dashboards

Projects

Express Tech

Excellence

Nuevo proyecto

Create Project

Select or drag and drop a file

Import

dispatchhealth

Create Project

New project

Start date

31/03/2023

End date

01/03/2023

Select your Project epics

☒ PART-234

☒ PART-289

☐ PART-306

Save

Welcome to ProjectNexus

localhost:3000/createProject

Actualizar

José

Team members

Projects

Dashboards

Projects

Express Tech

Excellence

Nuevo proyecto

Create Project

Select or drag and drop a file

Import

dispatchhealth

Create Project

Invalid time range

Name

Start date

dd/mm/aaaa

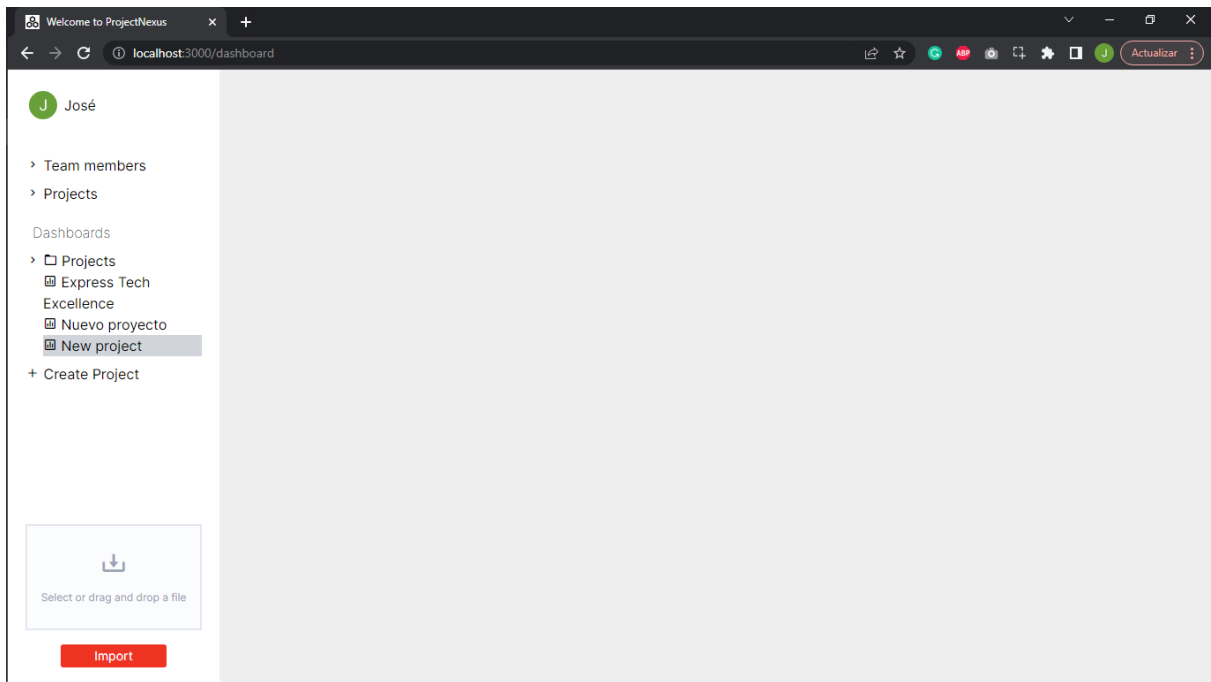
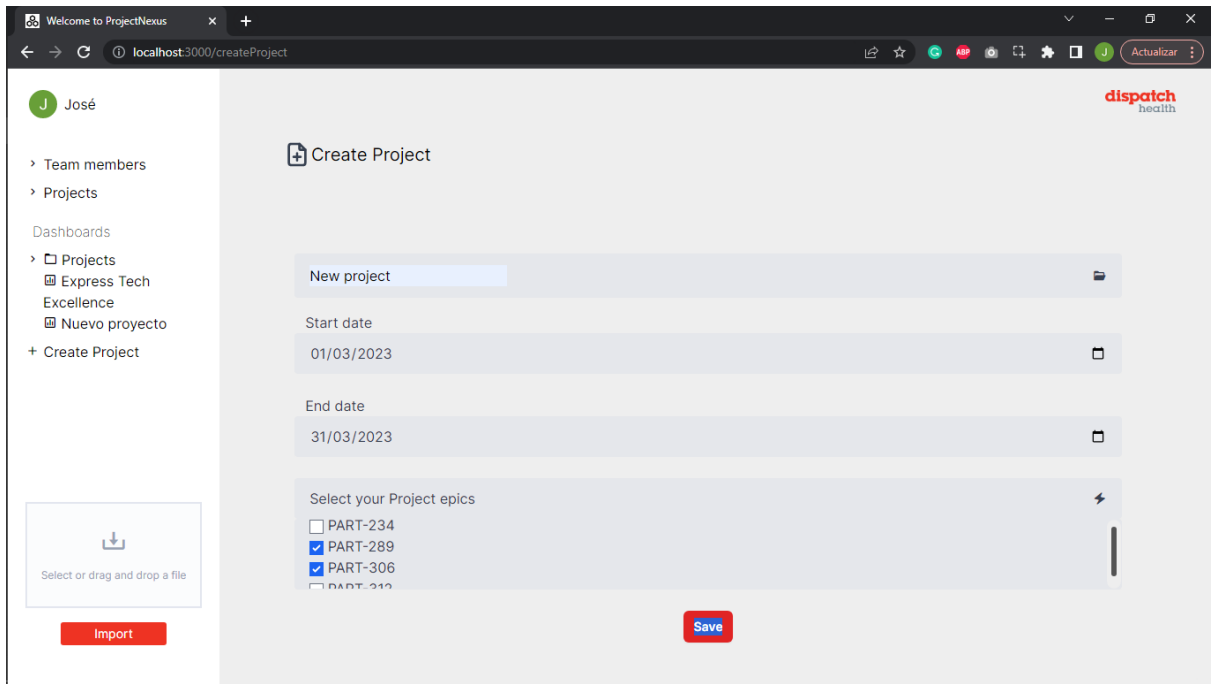
End date

dd/mm/aaaa

Select your Project epics

Save

Successful project creation



Authentication

1. Use case description:

ID: U.C. 03

Name: Authentication

Actor(s): User and Auth0

Description: This function allows unregistered users to get into the system to be able to use the functions that the system provides. The authentication is performed by Auth0 API.

Pre-conditions: Being unregistered and in the site URL.

Non-functional requirements: Having a @dispatchhealth.com email address, @tec.mx or @gmail.com



Events flow	
Normal Course	Alternatives
3.1 Clicks home link	
3.2 Clicks on “Log in with Gmail, dispatch health account or tec de monterrey account”	
3.3 System get Authentication Code Request	
3.4 API redirects to login prompt	
3.5 User authenticates and consent	3.5.1 If user does not agree to authenticate or the consent, return to 3.4
3.6 API gets Authentication Code	
3.7 System gets Authorization Code + Client ID + Client Secret	
3.8 API validates Authorization Code + Client ID + Client Secret	3.8.1 API negates access and redirects to login prompt (return to 3.4)
3.9 API brings ID Token and Access Token	
3.10 System Request user data with Access Token	
3.11 System responses	

Alternate Flows

A1→ If the email domain does not match Google, Dispatch Health or Tec de Monterrey domains, auth0’s API login window is prompted again.

A2→ If the user does not consent or agree to authenticate, auth0’s API login window is prompted again.

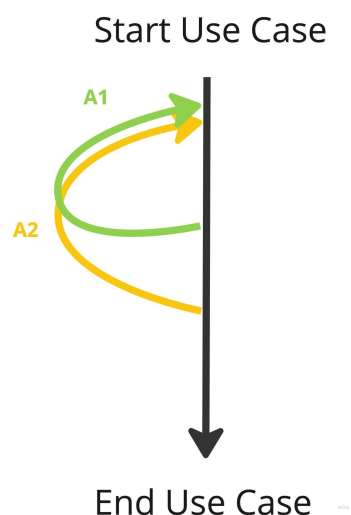


Table 1. Partial Scenario Matrix for the Authentication Use Case

Scenario Name	Starting Flow	Alternate
Scenario 1 - Email domain is invalid	Basic Flow	A1
Scenario 2 - User does not agree to authenticate or consent	Basic Flow	A2

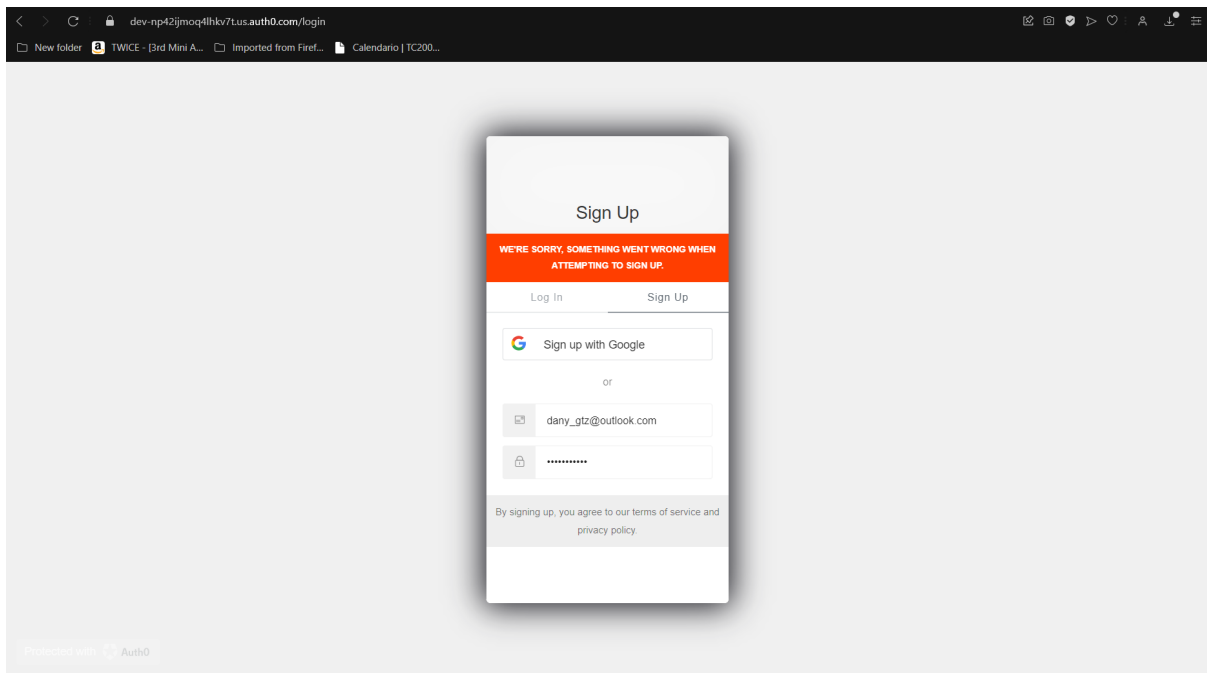
Table 2. Test case Matrix for the Authentication Use Case

Test Case ID	Scenario / Condition	User authenticated	Active Token	User's email domain has: @dispatchhealth, .com, @tec.mx, @gmail.com	User accepts authentication and consent with Auth0 system	User in database	Expected Result
TC 7.1	Scenario 1 - Email does not match valid domains	F	F	F	N/A	F	Redirects to login page
TC 7.2	Scenario 2 - User does not want to authenticate or consent	F	F	T	F	F	Redirects to login page
TC 7.3	Scenario 3 - User has been successfully authenticated and user already in database	T	T	T	T	T	Directly sends user to home page, user is authenticated and user's information is in database

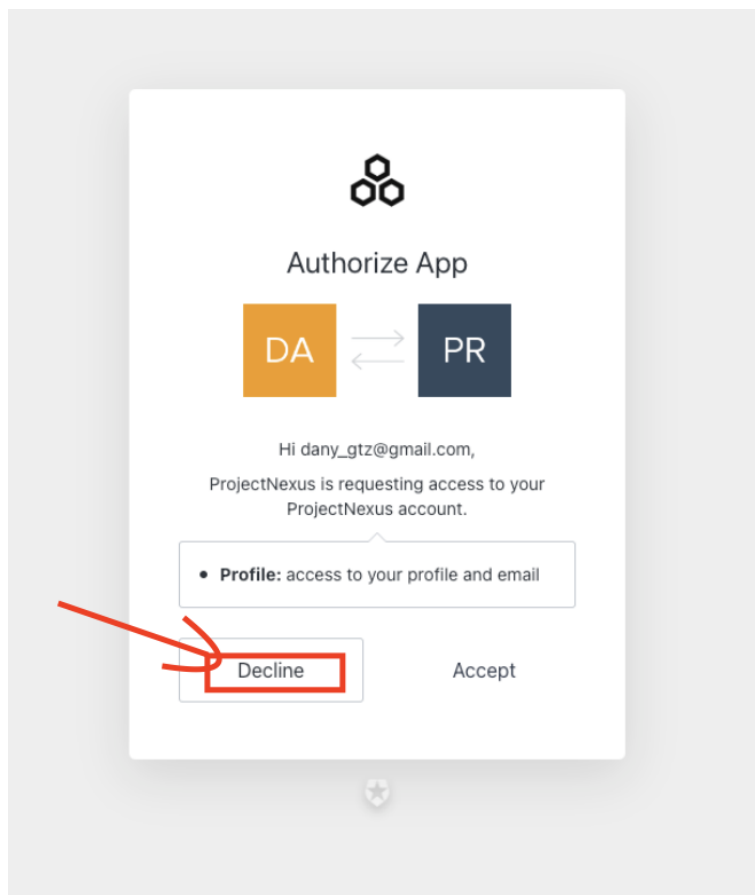
Table 3. Test Case Matrix with Data Values for User's authentication Use Case

Test Case ID	Scenario / Condition	Active Token	User's email domain has: @dispatchhealth.com, @tec.mx, @gmail.com	User accepts authentication and consent with Auth0 system	User in database	Expected Result
TC 7.1	Scenario 1 - Email does not match valid domains	N/A	dany_gtz@outlook.com	N/A	Insertion is not made	Redirects to login page
TC 7.2	Scenario 2 - User does not want to authenticate or consent	N/A	dany_gtz@gmail.com	No	Insertion is not made	Redirects to login page
TC 7.3	Scenario 3 - User has been successfully authenticated and user already in database	<random.length(32)> for 5 minutes	dany_gtz@gmail.com	Yes	User information has been inserted in database	Directly sends user to home page, user is authenticated and user's information is in database

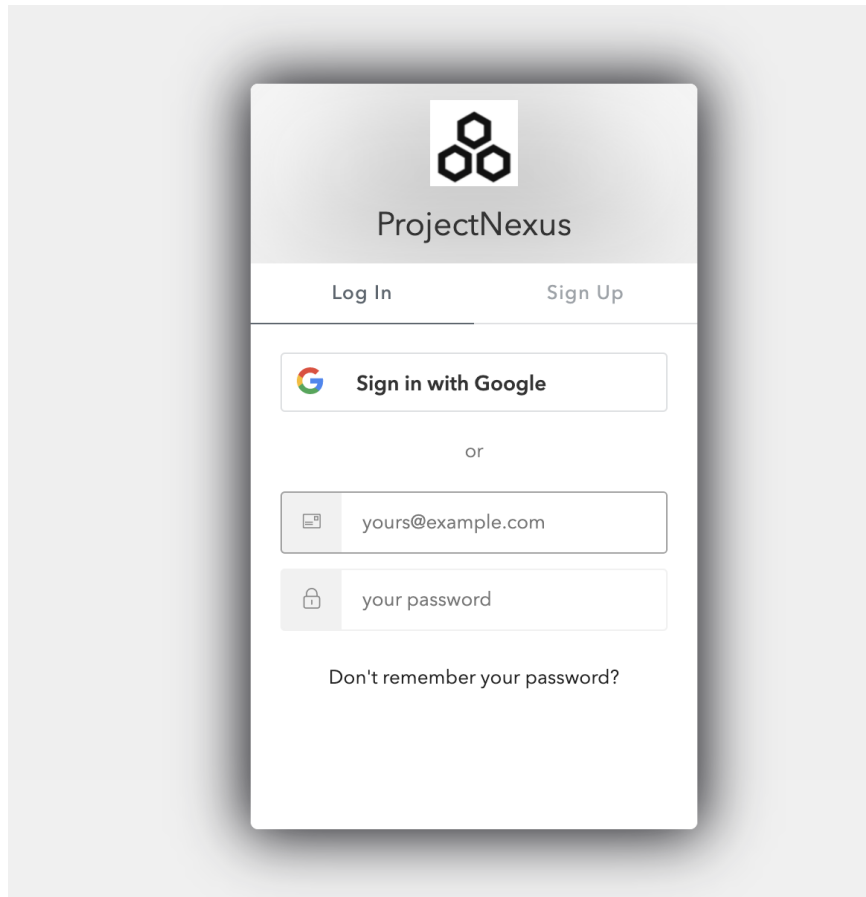
TC 7.1 Authentication error:



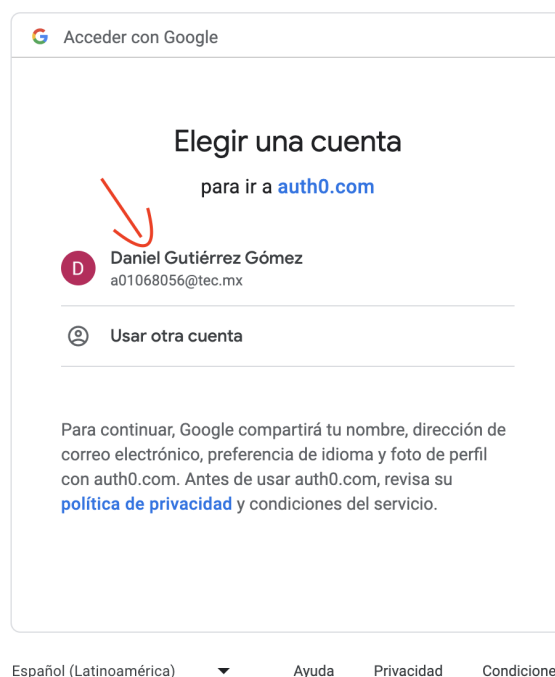
TC 7.2 User does not accept consent:



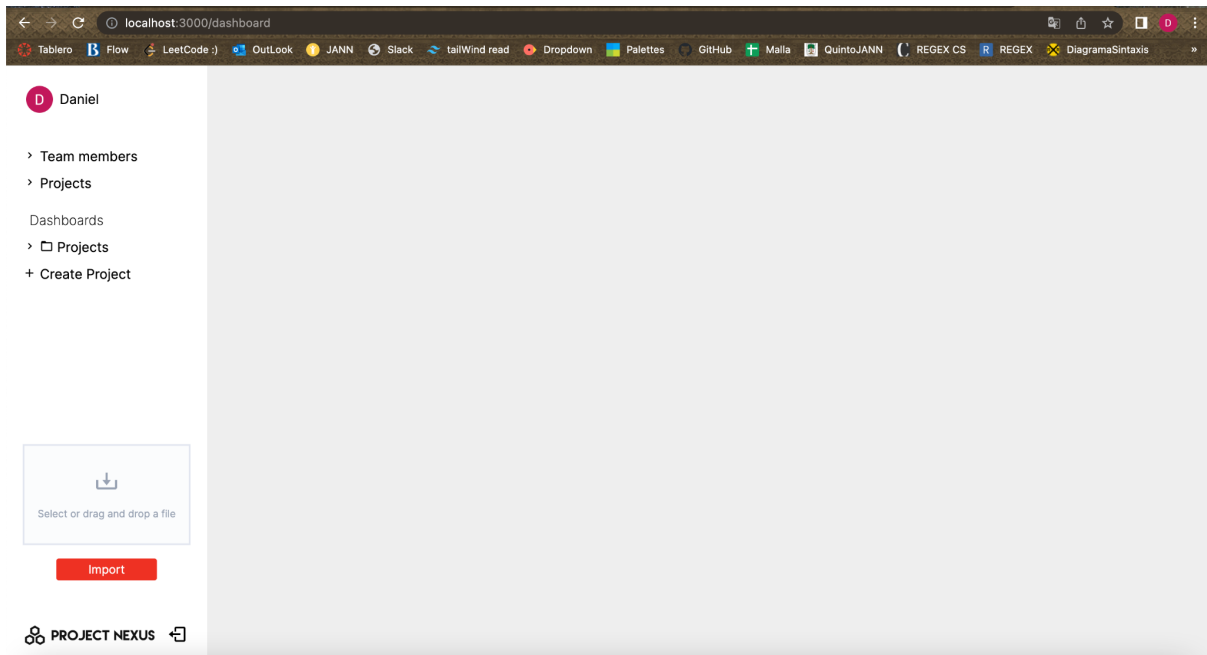
TC 7.2 User is redirected to login page when declining consent



TC 7.3 Successful authentication and login



User is send to main page



Logout:

1. Use case description:

ID: U.C. 08

Name: Logout

Actor(s): User and Auth0 system

Description: User can logout of our web application

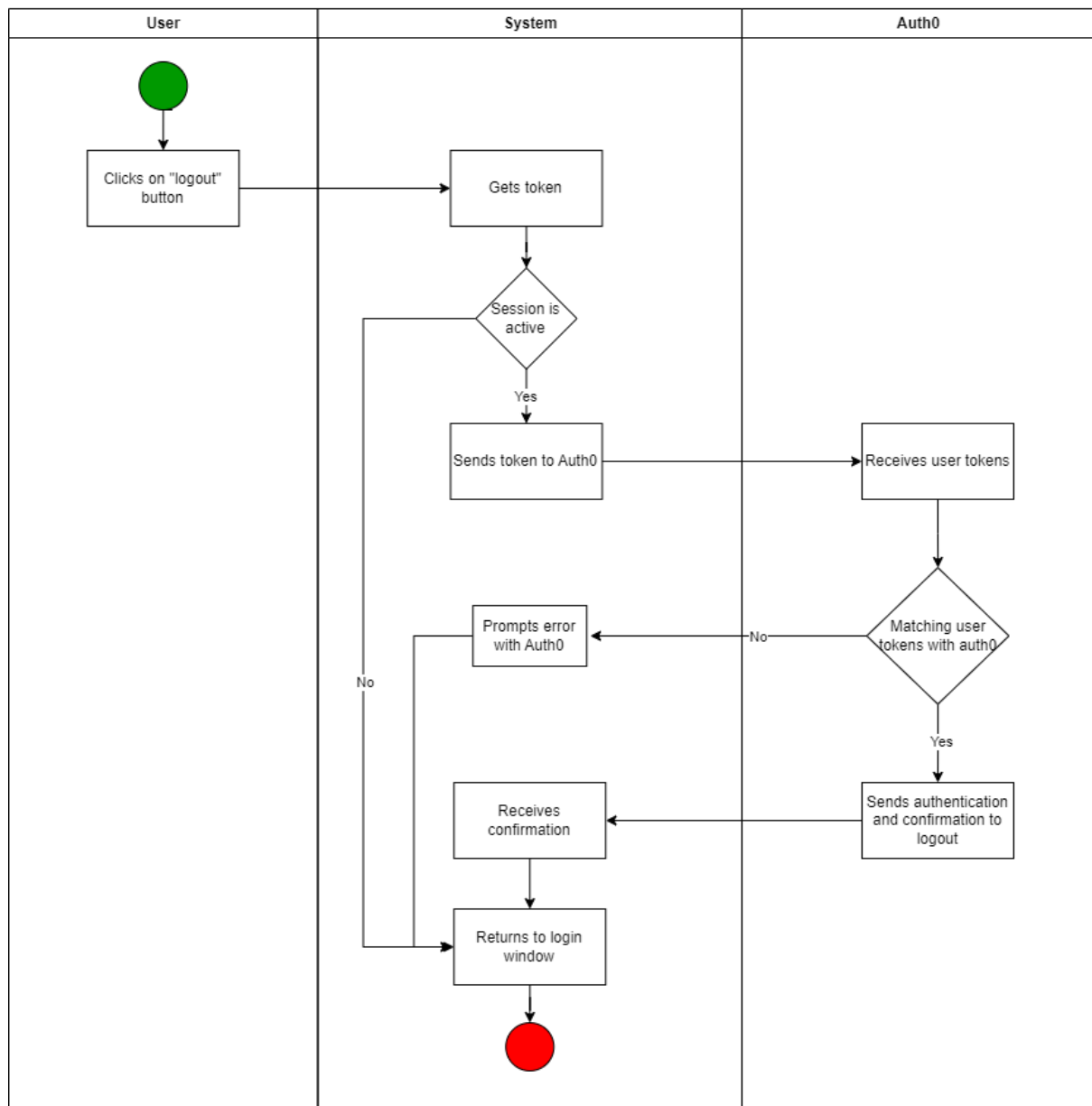
Pre-conditions: User has to be in an active session and has to be authenticated with Auth0

Post-conditions:

Non-functional requirements:

1. Security: User has to be authenticated by Auth0

Logout



Events flow	
Normal Course	Alternatives
8.1 User clicks on logout button	
8.2 System receives session tokens	
8.3 System validates if session is active	8.3.1 If not, jump to 8.9
8.4 System sends tokens to Auth0's system	
8.5 Auth0's system gets user tokens	
8.6 Auth0's system validates if user has	8.6.1 If not, system sends an error

been authenticated	message while making logout and jump to 8.9
8.7 Auth0's system sends to system validation of user's tokens and enables logout	
8.8 System receives validation	
8.9 System redirects user to login window	

Alternate Flows

A1→ If the session is not active, system redirects to login page

A2→ If Auth0's system does not validate if user has been authenticated, system redirects to login page

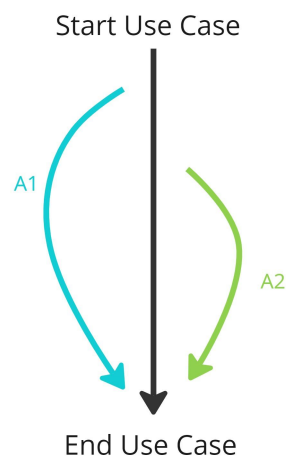


Table 1. Partial Scenario Matrix for the Logout Use Case

Scenario Name	Starting Flow	Alternate
Scenario 1 - Session is not active	Basic Flow	A1
Scenario 2 - System does not validate if user is authenticated	Basic Flow	A2

Table 2. Test Case Matrix for Logout

Test Case ID	Scenario / Condition	User authenticated	Active Token	System confirms authentication and with Auth0	Expected Result

TC 8.1	Scenario 1 - Session is not active	T	F	T	Redirects to login page
TC 8.2	Scenario 2 - System does not validate if user is authenticated	T	T	F	Redirects to login page
TC 8.3	Scenario 3 - Session is active and system validated user authentication	T	T	T	Redirects to login page

Log out



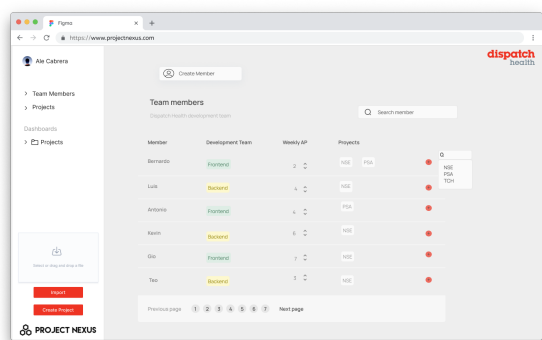
Usability

To test the usability and effectiveness of our app, we performed a think-aloud test with our client. We asked him to perform a series of tasks on our UI prototype and to verbalize his thoughts as he goes.

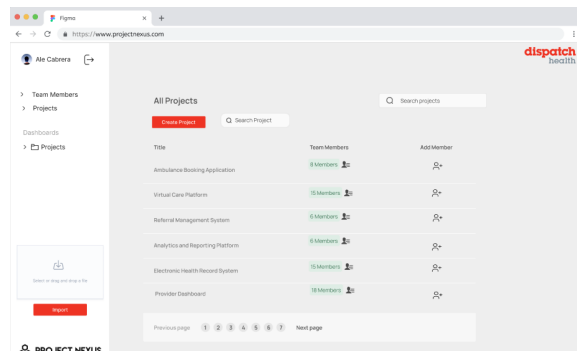
Think-aloud Test

With the feedback that we gathered, we were able to identify some usability issues, and we addressed them by revamping the following UI components with the goal of improving user experience. The changes will be made to ensure that our product is intuitive, user-friendly, and aligns with the expectations of our client's end-users.

Before



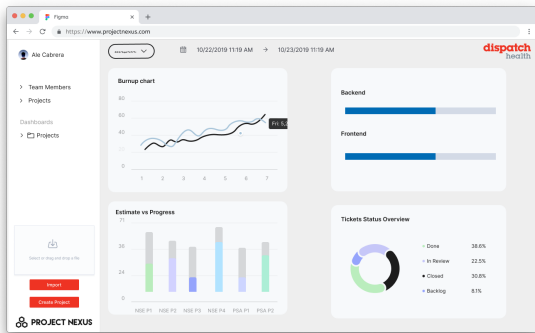
After



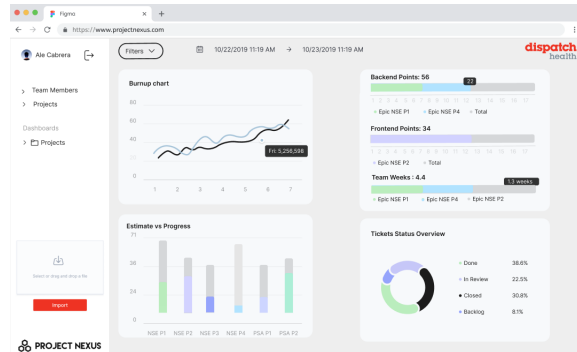
We've made significant improvements to the process of assigning Weekly Agile Points and Projects to Team Members. These functions can now be easily performed from the Projects tab, eliminating the need to switch back and forth between tabs.

Additionally, we've relocated the Create Project button to the Projects tab, further streamlining the workflow.

Before



After



We have enhanced the graph located on the top right of the Project Progress Dashboard. It is now more informative and descriptive as it breaks down the points by Epic, providing a clear overview of the project progress.

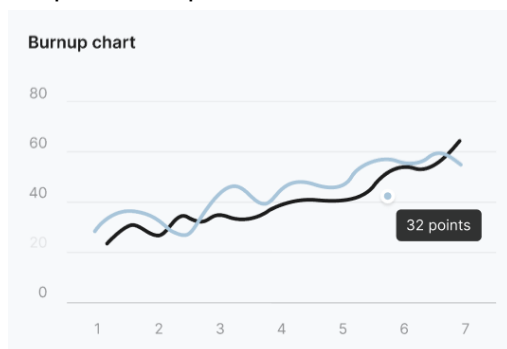
Moreover, we have included a new bar that displays the corresponding Team Weeks for each Epic, giving a better understanding of the team's workload and project timeline.

Reports

Title: Burnup Chart

Labels: Points hoover, weeks, points

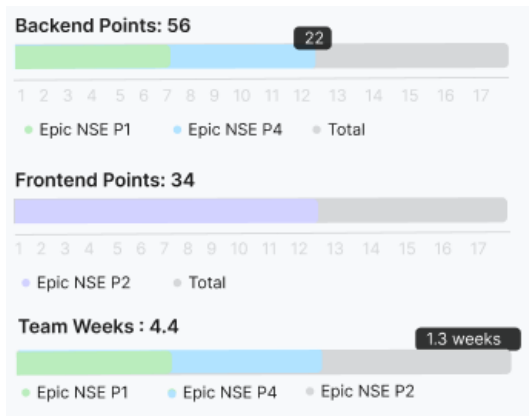
Graph: Burnup chart



Titles: Backend Points: #Points, Frontend Points: #Points and Team Weeks: #Weeks.

Labels: Epic link

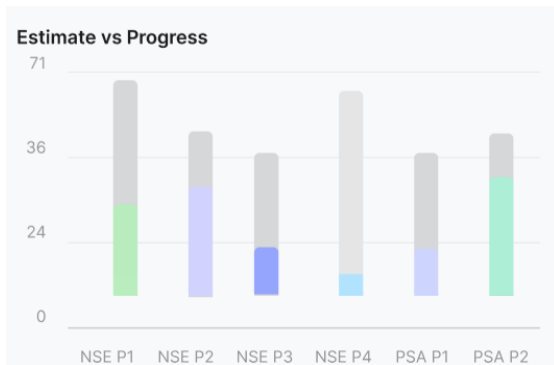
Graph: Progress vs Expected Plot Bar Graph



Title: Estimate vs. Progress

Labels: Epic link

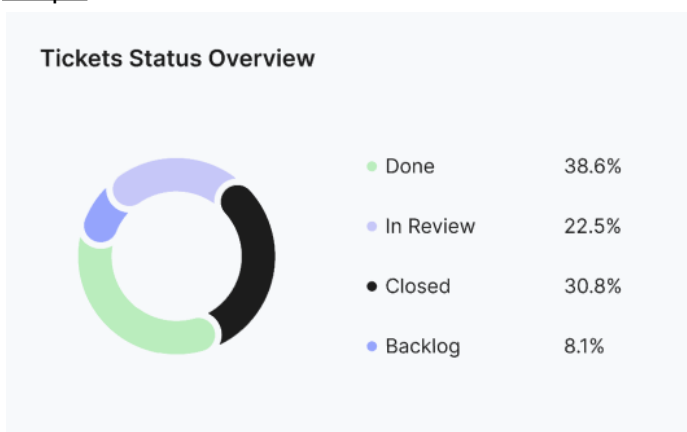
Graph: Progress vs expected points by epic in a Plot Bar Graph.



Title: Ticket Status Overview

Labels: Done, In review, Closed, Backlog, Percentage

Graph: Pie chart



Production Environment

To host the website, we will be using an AWS EC2 instance - a virtual server that provides scalable computing capacity in the cloud.

EC2 Benefits

- Security - the instances are located in a Virtual Private Cloud, which have built-in security features.
- Complete computing solution - EC2 works with other AWS components such as S3, DynamoDB, and Amazon SQS, providing a full computing, processing, and storage solution.
- Reliability - AWS guarantees a high level of availability and uptime.
- Cost-effectiveness - their on-demand instances are priced based on the resources that you use. You only pay for what you use, there aren't any upfront costs or long-term commitments.