Snapshot Week 05 of Group PG02

Project of ATSYS

No-Code Solution for InfluxDB

LeStartUP

Zilin Song - a1833935
Jen-Hao Liu - a1893169
Dang Quy Duong - a1893592
Baojing Li - a1894836
Shih-Han Lin - a1900715
Feinan Guo - a1903270
Xiaoqing Zhao - a1904344
Hao Jiang - a1907177
Ziqi Zhang - a1909438

Product Backlog and Task Board

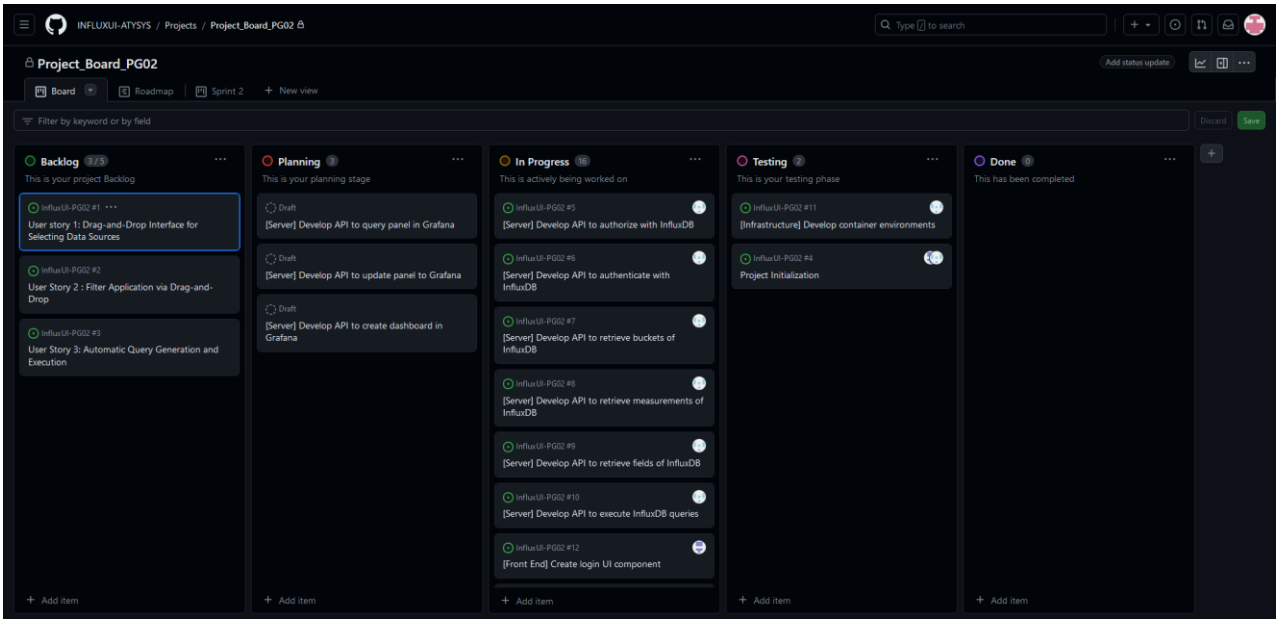
- The product backlog (continuous changes)

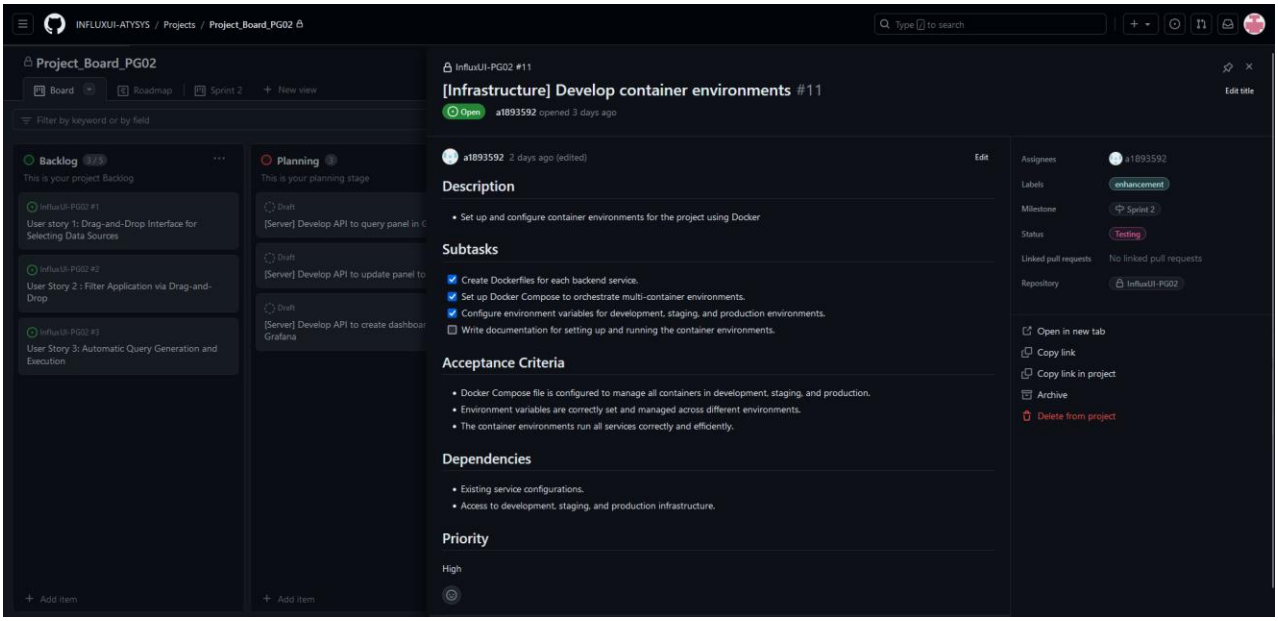
Category	Features	note
Front end	A single-page application using NextJS	
Front end	A login interface for user authentication at the same level of InfluxDB	
Front end	An intuitive drag-and-drop query builder for the Flux language	
Front end	Real-time Flux query generation	
Front end	Option to view the generated Flux query code	
Front end	Data visualization through native implemented charts and graphs	
Front end	Optional integration with Grafana dashboards and panels	
Back end	User authentication against InfluxDB	
Back end	Query validation and processing	
Back end	Data retrieval with InfluxDB	
Back end	Data processing for visualization	
Back end	Optional integration with Grafana dashboards and panels	
InfluxDB	Time-series database that powers the authentication of the web app and serves as the data source	
Grafana	Optional integration for saving and editing data queries and visualization dashboards.	

- The task board

Items	Tasks	Status
1	Software architecture	Version 1
2	Infrastructure for dev/staging/prod stages	On-going
3	Infrastructure for local InfluxDB, Grafana, Server and FE	On-going
4	From user story 1, form features + API of the app	On-going
5	Keep forming features and APIs	On-going
6	Develop BE using Django and APIs defined (Specifically query schema of IDB, query IDB, get/update/create Grafana panel	On-going

- The GitHub repository we are working on

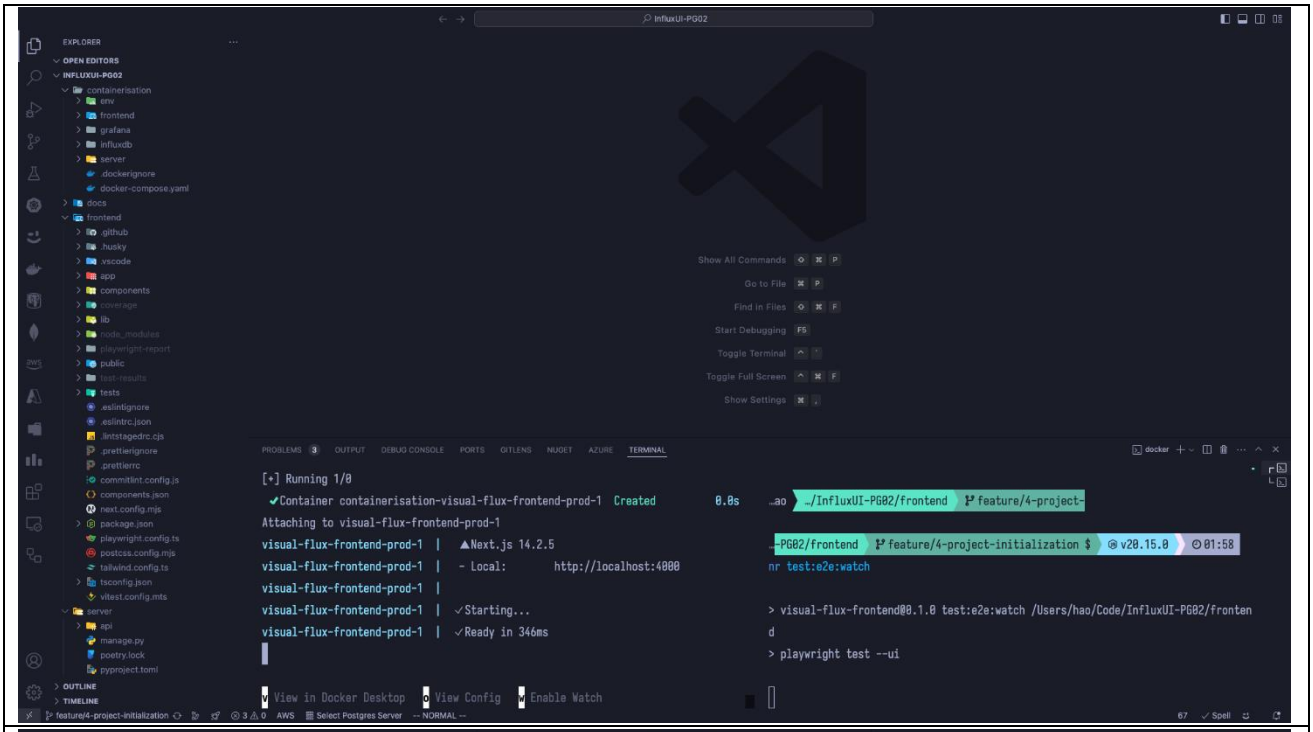


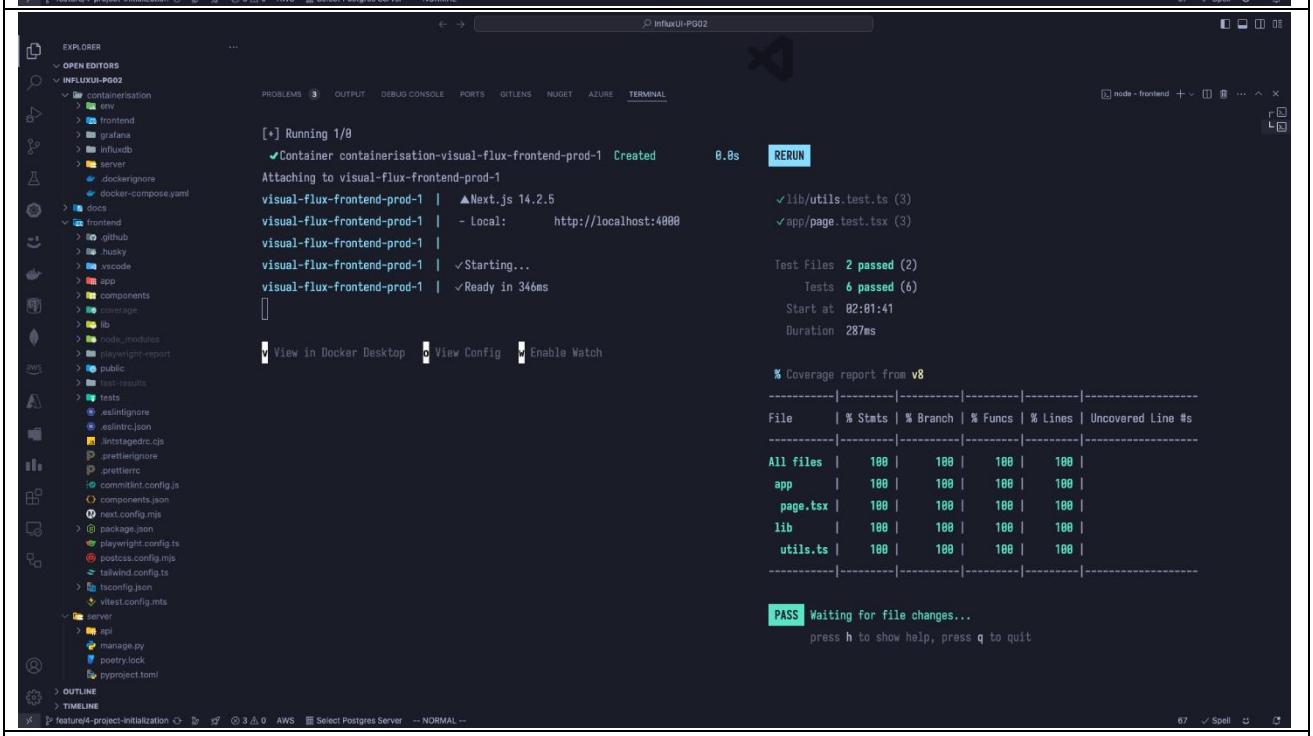


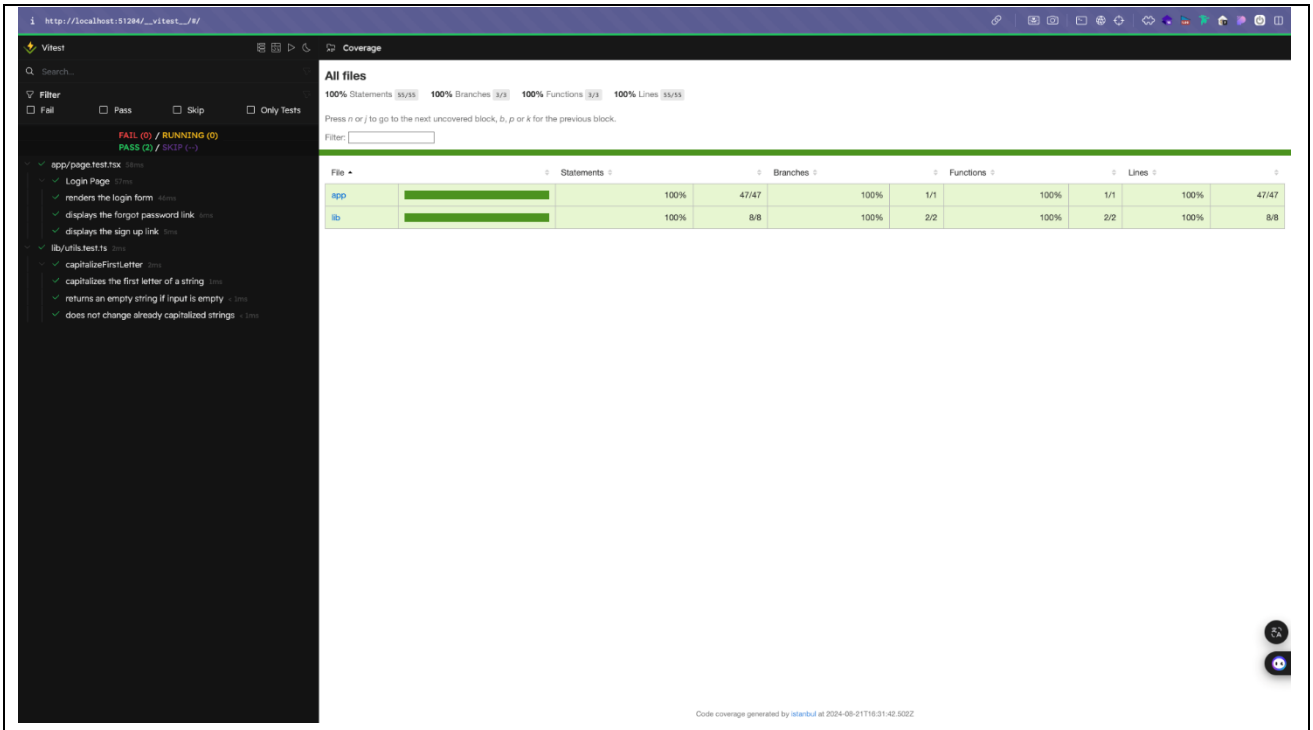
Sprint Backlog and User Stories

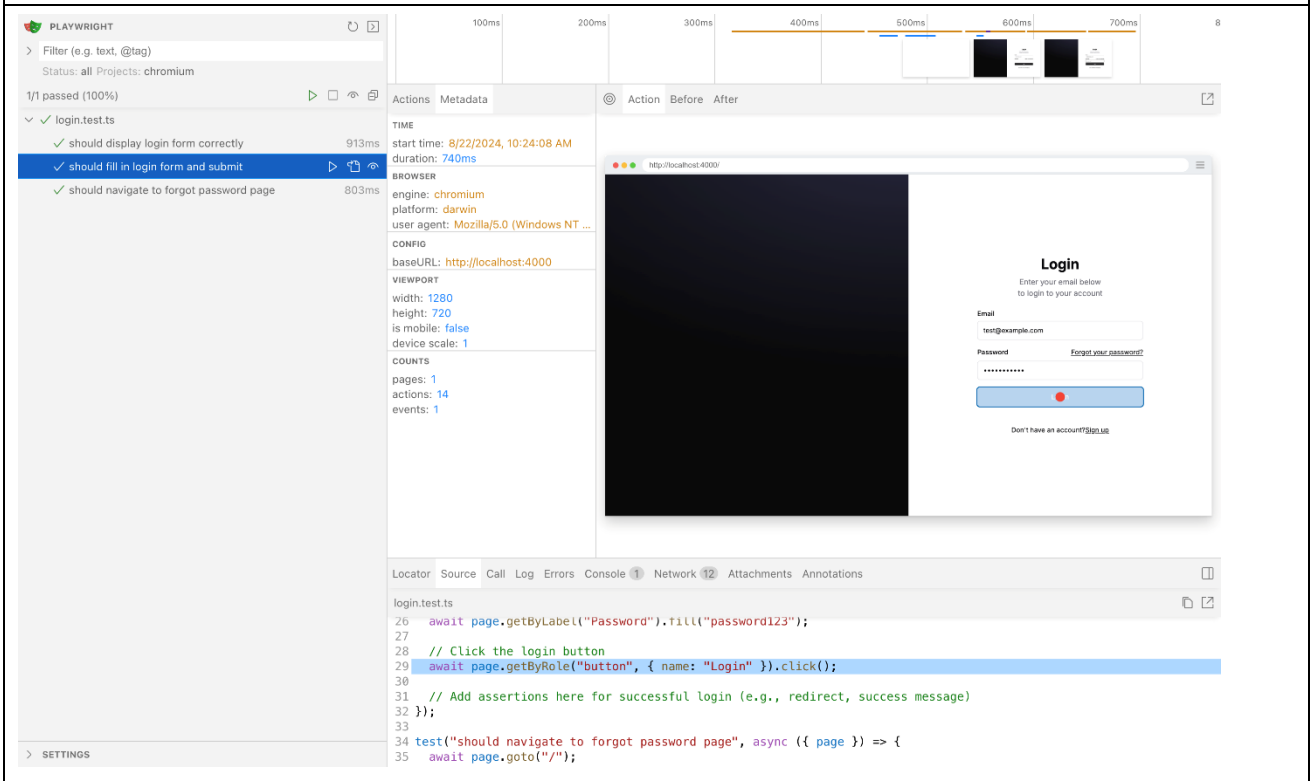
- The screenshot of the sprint backlog

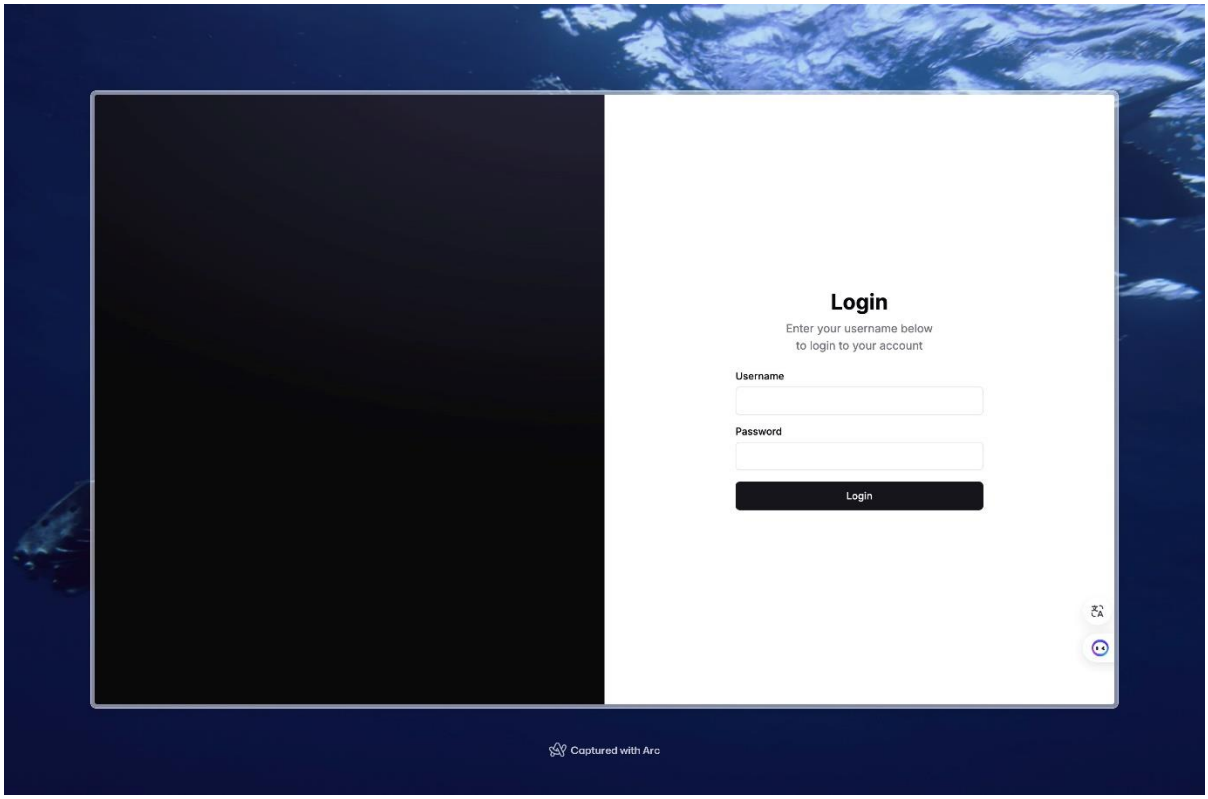
Items	Tasks	Status
1	Software architecture	Version 1
2	Infrastructure for dev/staging/prod stages	Done
3	Infrastructure for local InfluxDB, Grafana, Server and FE	Done
4	From user story 1, form features + API of the app	On-going
5	Keep forming features and APIs	On-going
6	Develop BE using Django and APIs defined (Specifically query schema of IDB, query IDB, get/update/create Grafana panel	On-going


InfluxUI-PG02
EXPLORER
OPEN EDITORS
INFLUXUI-PG02
Show All Commands
Go to File
Find in Files
Start Debugging
Toggle Terminal
Toggle Full Screen
Show Settings
PROBLEMS
OUTPUT
DEBUG CONSOLE
PORTS
GITLENS
NUGET
AZURE
TERMINAL
[+] Running 1/0
✔Container containerisation-visual-flux-frontend-prod-1 Created 0.0s
Attaching to visual-flux-frontend-prod-1
visual-flux-frontend-prod-1 | ▲Next.js 14.2.5
visual-flux-frontend-prod-1 | - Local: http://localhost:4000
visual-flux-frontend-prod-1 |
visual-flux-frontend-prod-1 | ✓Starting...
visual-flux-frontend-prod-1 | ✓Ready in 346ms
View in Docker Desktop
View Config
Enable Watch
nr test:e2e:watch
> visual-flux-frontend@0.1.0 test:e2e:watch /Users/hao/Code/InfluxUI-PG02/frontend
> playwright test --ui


InfluxUI-PG02
EXPLORER
OPEN EDITORS
INFLUXUI-PG02
PROBLEMS
OUTPUT
DEBUG CONSOLE
PORTS
GITLENS
NUGET
AZURE
TERMINAL
[+] Running 1/0
✔Container containerisation-visual-flux-frontend-prod-1 Created 0.0s
Attaching to visual-flux-frontend-prod-1
visual-flux-frontend-prod-1 | ▲Next.js 14.2.5
visual-flux-frontend-prod-1 | - Local: http://localhost:4000
visual-flux-frontend-prod-1 |
visual-flux-frontend-prod-1 | ✓Starting...
visual-flux-frontend-prod-1 | ✓Ready in 346ms
View in Docker Desktop
View Config
Enable Watch
RERUN
✓lib/utils.test.ts (3)
✓app/page.test.tsx (3)
Test Files 2 passed (2)
Tests 6 passed (6)
Start at 02:01:41
Duration 287ms
% Coverage report from v8
File | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
All files | 100 | 100 | 100 | 100 |
app | 100 | 100 | 100 | 100 |
page.tsx | 100 | 100 | 100 | 100 |
lib | 100 | 100 | 100 | 100 |
utils.ts | 100 | 100 | 100 | 100 |
PASS Waiting for file changes...
press h to show help, press q to quit


http://localhost:51204/__vitest__/#/
Vitest
Coverage
Search...
Filter
Fail
Pass
Skip
Only Tests
FAIL (0) / RUNNING (0)
PASS (2) / SKIP (--)
app/page.test.tsx
Login Page
renders the login form
displays the forgot password link
displays the sign up link
lib/utils.test.ts
capitalizeFirstLetter
capitalizes the first letter of a string
returns an empty string if input is empty
does not change already capitalized strings
All files
100% Statements 55/55
100% Branches 3/3
100% Functions 3/3
100% Lines 55/55
Press n or j to go to the next uncovered block, b, p or k for the previous block.
Filter:
File
Statements
Branches
Functions
Lines
app
100%
47/47
100%
1/1
100%
1/1
100%
47/47
lib
100%
8/8
100%
2/2
100%
2/2
100%
8/8
Code coverage generated by istanbul at 2024-08-21T16:31:42.502Z

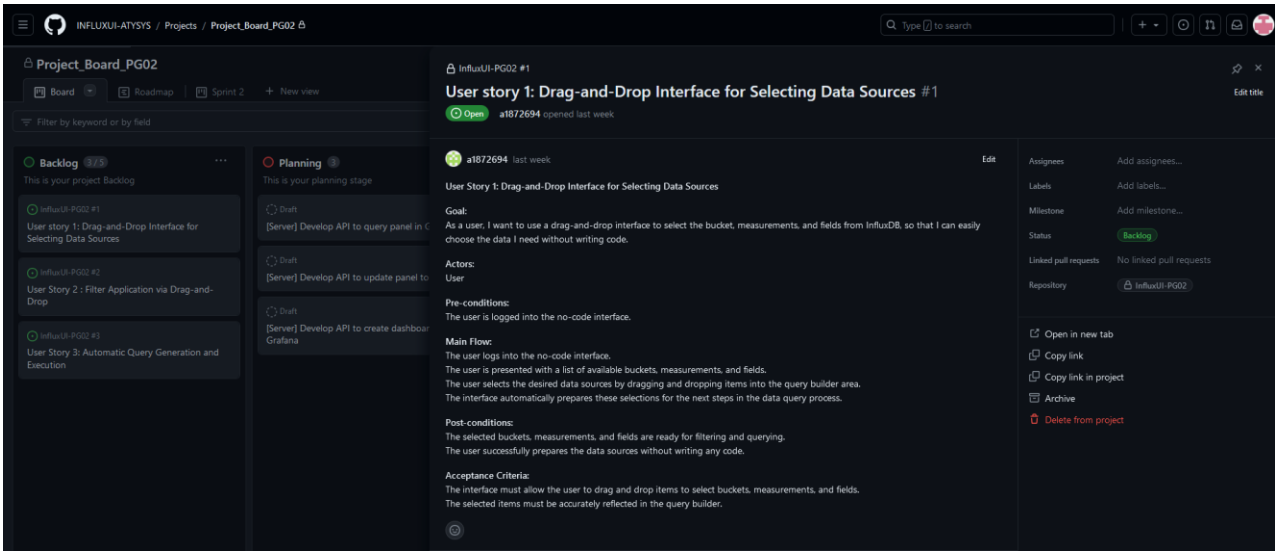

PLAYWRIGHT
Filter (e.g. text, @tag)
Status: all Projects: chromium
1/1 passed (100%)
login.test.ts
should display login form correctly 913ms
should fill in login form and submit
should navigate to forgot password page 803ms
SETTINGS
100ms
200ms
300ms
400ms
500ms
600ms
700ms
Actions
Metadata
TIME
start time: 8/22/2024, 10:24:08 AM
duration: 740ms
BROWSER
engine: chromium
platform: darwin
user agent: Mozilla/5.0 (Windows NT ...
CONFIG
baseURL: http://localhost:4000
VIEWPORT
width: 1280
height: 720
is mobile: false
device scale: 1
COUNTS
pages: 1
actions: 14
events: 1
Action
Before
After
http://localhost:4000/
Login
Enter your email below
to login to your account
Email
test@example.com
Password
Forgot your password?
Don't have an account? Sign up
Locator
Source
Call
Log
Errors
Console 1
Network 12
Attachments
Annotations
login.test.ts
26 await page.getByLabel("Password").fill("password123");
27
28 // Click the login button
29 await page.getByRole("button", { name: "Login" }).click();
30
31 // Add assertions here for successful login (e.g., redirect, success message)
32 });
33
34 test("should navigate to forgot password page", async ({ page }) => {
35 await page.goto("/");



- The user stories in the Sprint.
  - **User story 1: Drag-and-Drop Interface for Selecting Data Sources**

Goal	As a user, I want to use a drag-and-drop interface to select the bucket, measurements, and fields from InfluxDB, so that I can easily choose the data I need without writing code.
Actors	User
Pre-conditions	The user is logged into the no-code interface.
Main Flow	- The user logs into the no-code interface. - The user is presented with a list of available buckets, measurements, and fields. - The user selects the desired data sources by dragging and dropping items into the query builder area. - The interface automatically prepares these selections for the next steps in the data query process.

Post-conditions	- The selected buckets, measurements, and fields are ready for filtering and querying. - The user successfully prepares the data sources without writing any code.
Acceptance Criteria	- The interface must allow the user to drag and drop items to select buckets, measurements, and fields. - The selected items must be accurately reflected in the query builder.



Definition of Done

- Our current "definition of done":
  - Unit test passed.
  - End-to-end test passed.
  - Code reviewed in process: individual and group reviewed.
  - Non-functional requirements met. (If there is one)

Completed items

- In the 1st Sprint, our team had completed:
  - The team rules including hierarchy of periodic meetings and

communication platform.

- The team roles: Division of work including Scrum Master, front-end sub team and back-end sub team.
- The initial tech stack.
- Group development rules.
- Define the tasks of user story 1 on GitHub.
- The initial report which will be delivered to the client (Submission).

Meeting Minutes (in GitHub and Teams Files)

The 1st group meeting / The kick-off meeting 15:00-16:00, 2nd Aug 2024
The kickoff Sprint meeting / Q&A session with PO Sanchi Verma 15:00-16:00, 9th Aug 2024
The 1st Sprint meeting / Q&A session with PO Sanchi Verma 17:00-17:30, 14th Aug 2024
Meeting type: The 2nd group meeting 16:00-17:00, 15th Aug 2024
Meeting type: The 3rd group meeting 15:00-18:00, 23rd Aug 2024

Summary of Changes

In the first sprint, our team focused on establishing team rules, allocating roles, and laying the foundation for the development environment in accordance with the client's requirements. We successfully set up the development environment, including the front-end and back-end frameworks. The team was organized into specialized roles to enhance productivity, and responsibilities were clearly defined. Initial user stories were broken down into tasks, and we began work on implementing the core functionalities. We initiated the development process by creating the basic structure of the user interface, which will allow users to log in to the application.

This sprint primarily involved setting up the technical infrastructure and aligning the team to ensure a smooth development process in subsequent sprints. We

will continue to work on ensuring the integration with InfluxDB and Grafana for data visualization in future sprints.