**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 |
|  |  |  |

|  |
| --- |
|  |
|  |
|  |
|  |
|  |

* 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.