XState Builder

Team members: Sohum Dharamsi, Neil Randeri, Bryan Wheeler, Peter Dang | **Faculty adviser:** Irfan Ahmed, Ph.D. | **Sponsor:** Capital One | **Mentor:** Jacquelyn Dellinger

Problem

As a platform stakeholder, there is an increased need for workflow organization and management. However, there is currently no direct management or application for users to complete these requirements. This project aims to quickly and efficiently build a workflow description of our users' event-driven workflow process based on their requirements using XState. This application must be user friendly and be able to take in information from these users.

What is XState?

XState is a state management and orchestration solution for JavaScript and TypeScript apps.

- It uses event-driven programming, state machines, state charts, and the actor model to handle complex logic in predictable, robust, and visual ways.
- It provides a powerful and flexible way to manage application and workflow state by allowing developers to model logic as actors and state machines.

Objective

Develop an interactive and intuitive web application, backed by a database, utilizing XState to streamline the creation and management of event-driven workflow processes based on user-defined requirements. This application will display available XState definitions for users to select and allow them to trigger specific events for each definition. It will also enable functionalities like creating new instances, reviewing open instances, and managing task-based workflows. An intake form will guide users through creating customized workflow definition files by asking targeted questions. The form's responses will then be automatically committed to a GitHub repository. Our application will foster learning in areas such as XState, event-driven programming, front-end development, database management, and enterprise integration, ultimately enabling users to build and refine workflows with ease.

Approach

1. Build a **web application** that uses **XState** to process customer, event-driven actions taken by the user, using **Vue.js** and **Bootstrap**:
   a. Page: Landing Page (create new instance or review open instance)
   b. Page: Workflow Page (instance info, related tasks, select task, etc.)

↓

2. Create a **database** using **PostgreSQL** with **Instance** and **Task** tables to hold relevant data related to the information created using the web application:
   a. Table: Instance (has many tasks)
   b. Table: Task (has one instance)

↓

3. Create **intake form** to **collect input from users** and **output a definition file** for a specific workflow use-case:
   a. Definition files will be built using XState.

↓

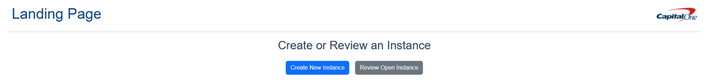
4. **Intake form** will ask **relevant, user-friendly questions** to build out the definition easily based on general user requirements. Sample questions:
   a. What is the instance type? What is the instance subtype?
   b. What is the flow for your definition?
   i. What is this state called?
   ii. What is the description of this state?
   iii. What event will your definition listen for in this state?
   iv. What action will need to be taken for this event?
   v. What is the target state for this event?

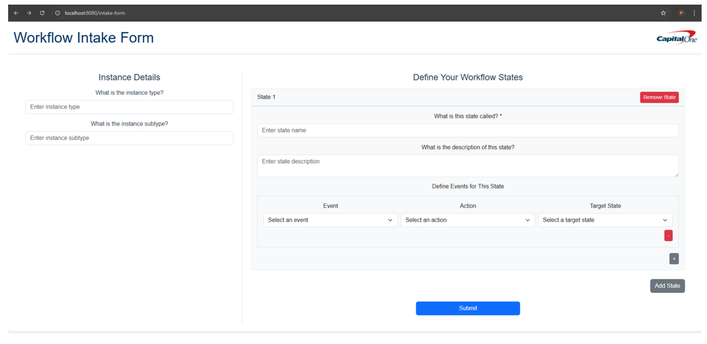
↓

5. **Intake form** should **open a PR** against a **GitHub repository** with relevant information filled out pertaining to the use-case, based on questions answered by the user.

Key Design Details

- Simple Docker container using vue.js.
- Landing page and Workflow page, both with buttons and dropdown menus.
- Vue.js and bootstrap libraries to implement an aesthetic and appealing user interface.
- Instance and Task tables within the database that are connected to the web application.
- User friendly intake form formatted with relevant questions.
- Separate pages to create or review an instance.





Vue.js XSTATE Bootstrap GitHub PostgreSQL

VCU College of Engineering

References

- 1. XState | Stately. Stately.ai. Published 2024. Accessed November 5, 2024. https://stately.ai/docs/xstate