Proceeding International Conference on Religion, Science and Education (2022) **1**, 651-655

Improving Effectiveness of State Management Using Prop Drilling Pattern on Jala Tech's Financial Feature

Albarra Naufala Erdanto*, Ari Sujarwo
Informatics Department, Faculty of Industrial Technology, Universitas Islam Indonesia,
Jl. Kaliurang KM. 14,5 Sleman Yogyakarta 55584, Indonesia. Tel. +62-274-895287, Fax. +62-274-895007.

Corresponding author*
18523158@students.uii.ac.id

Abstract: Jala Tech is a technology company that helps shrimp farmers in improving the quality of their ponds by utilizing web-based applications. One of the features of the Jala web application is the financial feature. State management in the frontend code of the feature uses the event bus pattern. However, there is a problem with implementing the event bus, namely the inability of Vue Devtools to debug the state. This inability makes it difficult for developers to track the flow of state changes in the application view. In addition, the decentralized method in storing application states causes state redundancy to occur in components that use the same state. Based on this, it is necessary to rewrite the code on the financial feature to change the state management from using the event bus pattern to the prop drilling pattern. The result of this research is a financial feature that has implemented prop drilling in the frontend code. By using prop drilling, it is possible to debug the state using Vue Devtools and centralize the states on the main parent component of the financial feature of the Jala web application. Therefore, developers can track state changes and more easily identify the state used in the financial feature of the Jala web application.

Keywords: Prop Drilling, State, Vue Devtools.

Introduction

In today's digital era, information technology is used by companies to increase the effectiveness of running business processes (Frestilia, 2013). One of the companies that utilize information technology is Jala Tech. Jala Tech is a company that helps shrimp farmers in increasing pond productivity by utilizing technology. One of the technologies used is a web-based application (Jala Tech, 2021).

Jala web application is developed using VueJS as a framework for the frontend side of the application. VueJS is a progressive framework for building web application interfaces. By using VueJS, it becomes easier to build an interactive application (Vuejs, 2021). VueJS also provides a tool called Vue Devtools to make it easier for VueJS developers to debug the application view. One of the advantages of using Vue Devtools is that it makes it easier for developers to monitor state activity in the application view (Cherckesova et al., 2021).

The state is data or variables contained in the application view that can change over time. The state is bound to the application view components so state activity causes changes to the application view. As the application grows, the state used becomes more and more complex. Therefore, good state management is needed according to application needs (Nelson, 2018).

Web components, also known as a components, are reusable application view components that can be used on various pages (Yang et al., 2002). Components can have states, functions, and props. Props are arguments passed from parent component to child component via component attribute (Wohlgethan, 2018). Like the state, props can affect the application view, but they cannot be changed directly by the component itself.

As of October 31, 2021, the Jala web application has ten active features (Jala Tech, 2021). One of the

ISSN 2829-3355 | EISSN 2828-8467

features of the Jala web application is a financial feature. The financial feature is a feature to record income and expenses and make financial reports for shrimp farmers.

The financial feature of the Jala web application uses the event bus pattern as a state management technique. An event bus is a medium that allows multiple components to send data to each other (Freeman, 2018). With the event bus, components can send data directly through the event bus entity as shown in Figure 1.

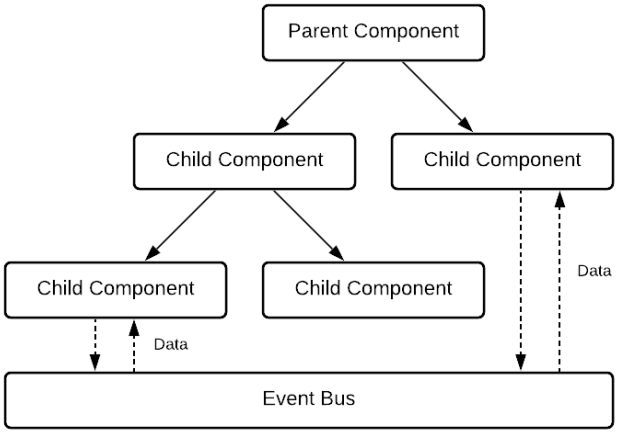

Figure 1. Illustration of data delivery using event bus.

The use of event bus patterns in the financial feature has several drawbacks. The first drawback is the inability to track the state event using Vue Devtools. This inability makes it difficult to investigate state activity, especially for new developers. The second drawback is the existence of state redundancy in each component that sends data to each other using the event bus. This is because an event bus is not a medium for storing the state, but a medium for sending data. The state that was successfully sent by the component via the event bus must be stored as the same state as the receiving component. Therefore, some states that have the same function can encounter state duplication. This can be a problem if the same states are out of sync with each other and have different values.

Based on the problems above, this study is to increase the effectiveness of state management by rewriting the state management from using an event bus to a prop drilling pattern on the financial feature of the Jala web application. Improving the effectiveness of state management is by providing the ability to debug the state using Vue Devtools and centralize the states on the parent component of the Jala web application feature so that developers can easily identify the state used for the feature.

Materials and Methods

Scrum

The application development method used at Jala Tech is Scrum. The scrum method is one of the agile software development methods and is very effective in handling complex projects with tight schedules (Pressman, 2010). There are several stages in software development using Scrum at Jala Tech, namely sprint planning, development, sprint review, and sprint retrospective.

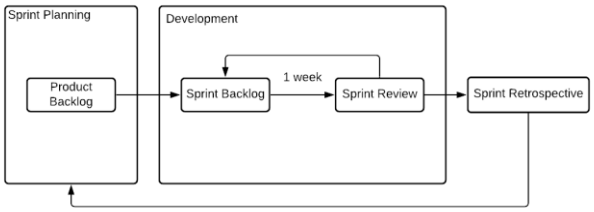

Figure 2. Software development flow using scrum.

The first stage is sprint planning. Based on Figure 2, sprint planning consists of defining a product backlog which is then broken down into several sprint backlogs. The sprint backlog is a list of tasks that the developer needs to do in each sprint. At Jala Tech, each sprint is done within one week. After doing sprint planning, the next stage is development. In the development stage, developers work on tasks according to the previously planned sprint backlog. At the end of each sprint, there is a sprint review to check the work that has been done and communicate updates and problems. The final stage is the sprint retrospective. The sprint retrospective is an evaluation stage of all sprints that have been carried out which aims to improve team performance in further development.

Prop Drilling Pattern

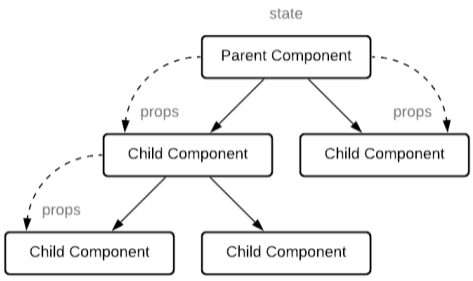

Figure 3. Illustration of data delivery using prop drilling

Based on the problems in the introduction, the use of the event bus has some drawbacks in financial features. In this study, prop drilling is used to replace the use of the event bus. It aims to increase the effectiveness of state management. Prop drilling pattern is a state management pattern that transmits data using props between parent and child components (Kankaala, 2019). In the prop drilling pattern, all the states of the financial feature are stored in the parent or main component. If a child component or even a deeper component wants to use state, the parent component needs to send the state sequentially from the outermost component using props as can be seen in Figure 3. In the process of changing the state management pattern, code rewriting was carried out on all frontend code in the financial feature of the Jala web application.

Results and Discussion

The financial feature that has used prop drilling pattern

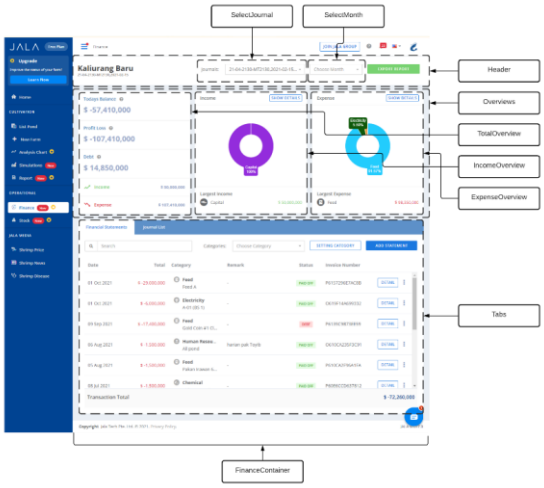

Figure 4. Interface design and component structure on financial feature

In this study, the results were obtained in the form of frontend development on the financial feature of the Jala web application that applies prop drilling patterns as state management. The structure of the components in this feature can be seen in Figure 4. By using prop drilling, all states of the feature are stored in the parent component as shown in Figure 5. The parent component needs to pass the states via props if the child component needs data from that state. As shown in Figure 6, the parent component in the financial feature is named FinanceContainer. FinanceContainer component has three child components, namely Header, Overviews, and Tabs. In addition, the Header component also has two child components, namely SelectJournal and SelectMonth and Overviews Component has three child components, namely TotalOverview, IncomeOverview, and ExpenseOverview.

```
export default {
  name: 'FinanceContainer',
  data () {
    return {
      profile: null,
      journal: null,
      startDate: '',
      journalsPaginated: {},
      financesPaginated: {},
      financesTotal: 0,
      currentCash: 0,
      currentDebit: 0,
      currentCredit: 0,
      currentProfit: 0,
      currentDebt: 0,
      incomes: [],
      expenses: [],
      financeKeyword: '',
      journalKeyword: '',
      financeCategoryId: '',
    }
  },
}
```

Figure 5. States used in financial feature

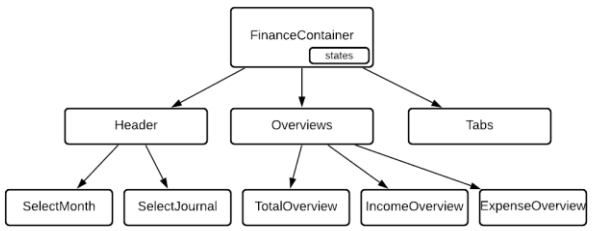

Figure 6. Component hierarchy on financial feature

The pattern of sending state data using prop drilling is carried out sequentially from the outermost component. As in Figure 6, the SelectJournal component is a child component of Header and a grandchild component of

FinanceContainer. Therefore, if the SelectJournal component requires states, FinanceContainer needs to send the state to Header first, then Header sends the state to Select Journal. An example of a data transfer code using props in VueJS is shown in Figure 7.

Figure 7. Use of props on VueJS components

Ability to debug the states via Vue Devtools

The state mutation flow can be tracked using Vue Devtools. Based on Figure 8, state events are recorded in Vue Devtools. The data recorded in Vue Devtools includes the name of the event, the component that performed it, and the time of the event. The payload of the data sent in the event is also recorded in Vue Devtools.

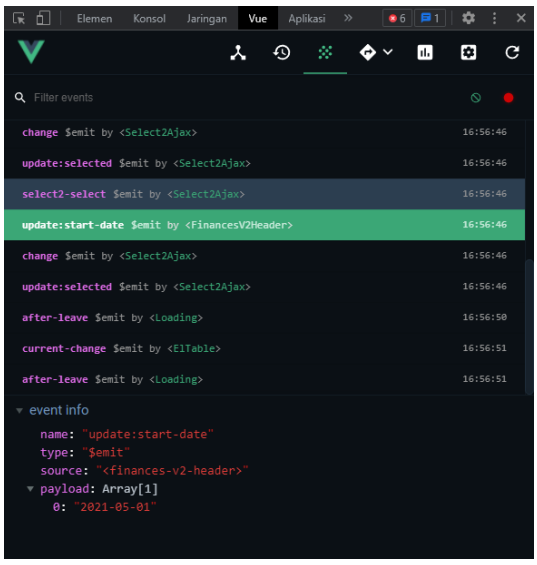

Figure 8. Use of props on components

Discussion

Based on the results of an interview from the Lead of Software Engineer, Farid Inawan, said that centralized the states in the parent component of the financial feature make it easier for developers to identify the state used in this feature. Therefore, the development that is carried out is faster because the developer immediately knows the states. In addition, it also minimizes the existence of redundancy in the states. The ability to track state mutations using Vue Devtools also makes it easier for developers to identify state behavior that does not meet expectations, thereby speeding up the debugging process.

Conclusions

Based on the results of applying the prop drilling pattern to the financial features of the Jala web application, state management using prop drilling is more effective because of the ability to track events in the state using Vue Devtools, thereby speeding up the error investigation process in the development of the Jala web application frontend. In addition, centralized state storage makes it easier for developers to identify the state used in financial features and minimizes state redundancy.

References

Cherckesova, L., Boldyrikhin, N., Revyakina, E., Safaryan, O., & Yengibaryan, I. (2021). Development of a real-time document approval system. In E3S Web of Conferences (Vol. 273, p. 08047). EDP Sciences.

Frestilia, N. (2013). Pengaruh Pemanfaatan Teknologi Informasi, Karakteristik Informasi Sistem Akuntansi Manajemen, Dan Ketidakpastian Lingkungan Terhadap Kinerja Manajerial. Jurnal akuntansi, 1(1).

Freeman, A. (2018). Loosely Coupled Components. In Pro Vue. js 2 (pp. 451-477). Apress, Berkeley, CA.

Jala Tech. (2021). Jala: Solusi Digital untuk Kesuksesan Tambak Udang Anda. https://jala.tech

Jala Tech. (2021). Jala Tech Web App. https://app.jala.tech Vuejs. (2021). Introduction. https://vuejs.org/v2/guide/index.html

Kankaala, M. (2019). Enhancing E-Commerce with Modern Web Technologies.

Nelson, B. (2018). State Management. In Getting to Know Vue. js (pp. 127-147). Apress, Berkeley, CA.

Pressman, R. S. (2005). Software engineering: a practitioner's approach. PalgraveMacmillann.

Wohlgethan, E. (2018). SupportingWeb Development Decisions by Comparing Three Major JavaScript Frameworks: Angular, React and Vue. js (Doctoral

dissertation, Hochschule für Angewandte Wissenschaften Hamburg).

Yang, J., & Papazoglou, M. P. (2002, May). Web component: A substrate for web service reuse and composition. In International Conference on Advanced Information Systems Engineering (pp. 21-36). Springer, Berlin, Heidelberg.