Lucia

JavaScript Library for Developing Performance-Focused Web Applications

Aiden Bai
Old Websites
- Simple and static
- Non-interactive

New Websites
- Interactive
- Slow and complex
- Bad user & developer experience

Major issue for web developers
Design Principles

- **Speed**
  - Conservative execution
  - Immutable patterns
  - Lightweight core

- **Simplicity**
  - Scalability
  - Reactivity

Figure 2: Spreadsheet reactivity dependency example (Bai)
Methods & Procedure

- **Language**: TypeScript
- **Build**: Rollup/Babel
- **Repository**: Github
- **Publish**: NPM + Unpkg
- **Testing**: Jest, Ad-hoc
- **Code Editor**: VSCode

![Flowchart](Figure 3: Library creation procedure flowchart (Bai))
Figure 4: Lucia Core process flowchart (Bai)
Results & Analysis

Figure 5: Comparison table of major JavaScript Libraries (Bai)

Figure 6: Column Graph of weighted score for JavaScript Libraries (Bai)
Conclusion & Next Steps

- Simple to integrate and super fast
- Implemented in many use cases - production ready
- Learned about JavaScript library internals - 8 months of work
- Innovated upon pitfalls for a more useful product

Next Steps:
- Promote to more developers
- Create tooling for 1.0.0
References