

Interactive Evaluation of Shortest Path Methods

sddec23-14

Alex Blomquist, Selma Saric,

Samuel Caldwell, and Yadiel Johnson

Design & Broader Context

Areas Summary:

- Public health/Safety/Welfare
 - Reduce CO2
- Global/Cultural/Social
 - Reduction in travel time
- Environmental
 - Improved code efficiency
- Economic
 - User product self-improvement

Design & Broader Context (cont.)

Prior Work/Solutions

Advantages

- Evaluate complex data sets
- Provide empirical data
- Allows direct comparison between algorithms

Disadvantages

- Less detailed visualization
- Not intended for users with no SSSP algorithm experience.

User Needs

- Educators and Students
- Researchers

Design & Broader Context (cont.)

Technical Complexity

- The system as a whole should support multiple algorithms and datasets.
 - Scalable format for adding algorithms and datasets
 - Data persistence
- The system must use statistical analysis to gather and present highly accurate algorithm information to users.
- Parallel execution of algorithms to fulfill user requests.

Design Exploration

- **Design Decisions**
 - Web Application
 - Algorithm Scope
 - Server & Algorithm Driver
- **Decision-Making & Trade-Offs**
 - UI Decision Matrix

Criteria	Weight	CSS	HTML	JavaScript frameworks	Bootstrap	Foundation
Ease of use	3 *	3	4	3	4.5	3.5
Documentation	4 *	4	5	2	4	3
Learning Curve	4 *	4	5	2	4	3
Performance	5 *	5	5	3	4	4
Customization	5 *	5	4	5	4	4
Total	21	76	64	45	62.5	49.5

Visualize Output of Algorithm	Clean, Easy-to-Understand Visualization	Have Multiple Visualizations If User Selects Multiple Algorithms
Visualization is Based On User's Selected Algorithm(s) and Data Set	A Visualize Algorithms	Have Multiple Visualizations If User Selects Multiple Data Sets
Algorithm Visualizer Connects to Algorithm Execution Driver in Backend	Visualizes Shortest Path Using Selected Algorithm	Colorful and Appealing Visualizations

User Can Upload One or Multiple Data Sets	User's Data Set(s) Will Be Run Through the Different Algorithm(s) They Choose	Data Set Must Be In a Certain File Format To Work on the Web App
The App Will Be Able To Handle Small and Large Data Sets	B Allow User to Upload Data Set	Report Will Be Generated After The User's Data Set Runs Showing Different Algorithm Output Metrics
User's Data Set Will Be Stored in Server	Shortest Path Will Be Calculated Using The Data Set and The Algorithm(s) The User Chooses	There Will Be a Cap On How Large The Data Set Can Be

Use Algorithm Research From Taxonomy Document	User Can Pick Multiple Algorithms to Run	User Can Choose From a List of Different Algorithms
User's Algorithm Output Metrics Will Be Stored in a Report	C Allow User to Select Algorithm	User Can Run One Algorithm on One or Multiple Data Sets
User's Algorithm Selection Connects to Backend Algorithm Execution Driver	Output of Algorithm Selected By User Will Be Visualized	User Can Run Multiple Algorithms On One or Multiple Data Sets

After an Algorithm Runs, Report Will Be Generated	Report Will Detail Various Metrics on the Algorithm's Output	Report Will Get Stored On The ETG Server
Algorithm Comparison Results Will Be On Report	D Generate and Store Report	New Report Gets Generated After Every Execution

A Visualize Algorithms	B Allow User to Upload Data Set	C Allow User to Select Algorithm
D Generate and Store Report	Interactive Evaluation of Shortest Path Methods	E Algorithm Comparison
F Good User Interface	G ETG Server	H Algorithm Execution Driver

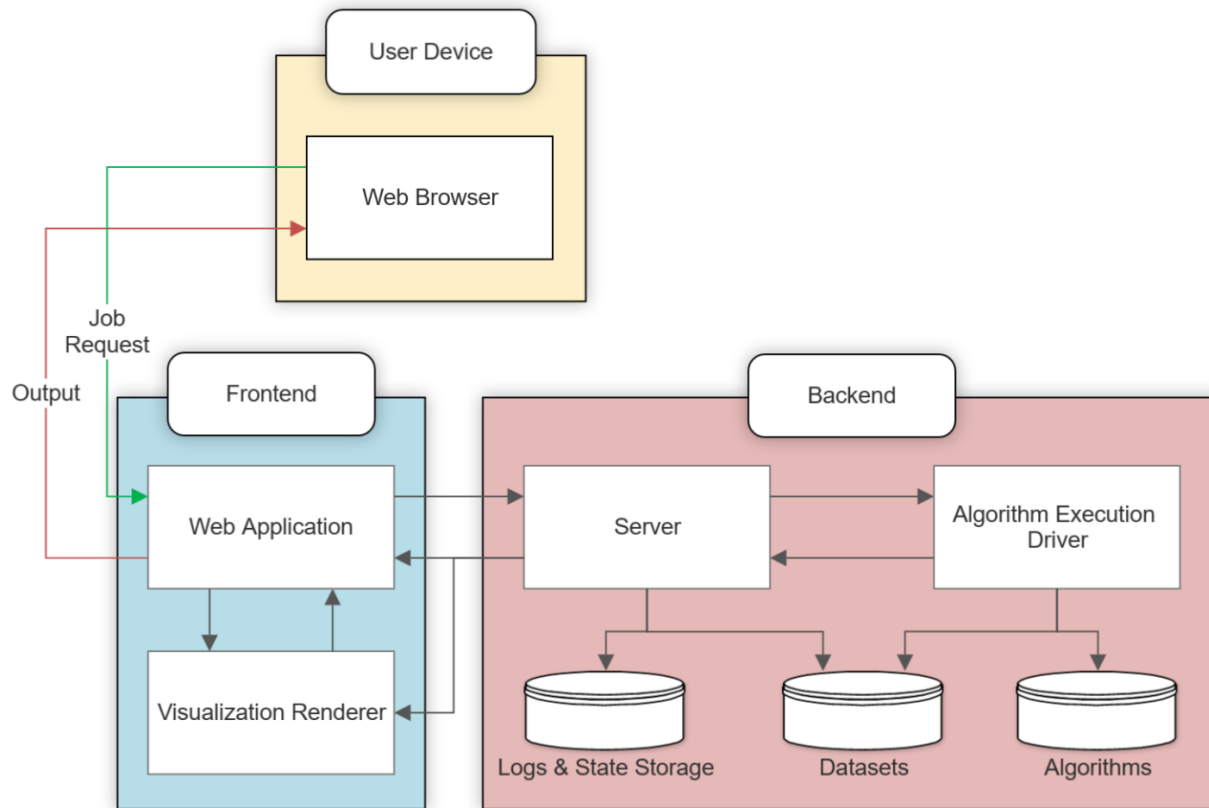
Algorithms Can Be Compared To Find Most Efficient One for User's Data Set	Results Get Stored In Report	Comparison Results Will Be Shown After User's Selected Algorithm(s) and Data Set(s) Run
Various Algorithm Visualizations Will Be Compared	E Algorithm Comparison	Algorithm Runtimes Will Be Compared

Clean and Organized UI to Make Navigating the Web App Easy	Colorful UI to Make the Web App Look More Visually Pleasing	Have a Drop Down List for User's To Select Which Algorithm(s) They Want to Run
Integrate Bootstrap So Web App Has Easy-To-Use UI on All Devices	F Good User Interface	Organized UI for Algorithm Comparison Results So User Can Easily Understand the Outputs
Format UI With HTML and CSS	Create User Centered Design for UI	Ensure that UI is Not Cluttered and That Everything is Evenly Spaced For Readability

Need to Install VM to Access Server	Store the Scripts for Selecting Algorithms	Store Data Sets Uploaded By User
Server Needed for Web App to Handle Multiple Algorithm Executions	G ETG Server	Need to Request To Use and Access a Server For Our Project
Stores Generated Algorithm Output Reports	Communicates with Algorithm Execution Driver in Backend	Request Access Before Second Semester of Senior Design

Holds the Backend Algorithm Execution	Connects to Frontend So It Can Display The Algorithm Output Results	Completes The Algorithm Computation On The User's Data Set
Communicates with Server in Backend	H Algorithm Execution Driver	Algorithm Output Results That It Calculates Get Sent To Visualization Renderer to Be Mapped

Proposed Design



Functionality

- Traditional Model-View-Controller full stack application

Intended for...

- Product Development
- Academic Research
- Educational Resource

Areas of Concern & Development

- Accuracy and correctness of information