# What?

Online finite state machine graphing tool

# Why?

To provide developers with a tool that they can use on any platform and be able to access their work from anywhere. When it’s completed (graphing portion), it will be more user friendly than other products out there (i.e., Visio) because it has one purpose, to design state machines.

# Technologies used

1. JavaScript (jQuery and Bootstrap libraries implemented)
2. HTML
3. CSS
4. Server Platform: Node.js
5. Database: MongoDB (Document Oriented Database. I.e., JSON format)

# Why these technologies?

I have a background as a JavaScript developer and kept hearing about Node.js…so I wanted to give it a try. Also, after further research, using a full JavaScript stack for client-side and server side seemed like it would save me from the aggravation of context switches between languages as well as converting data from JSON to another format.

# Design Pattern

MVC

# Demo:

1. User Interface
   1. Navigation Bar
      1. Action Menu
      2. Account Menu
   2. Instruction Panel
   3. Canvas
2. Login Success
3. Login Fail
4. Logout
5. Create Graph
6. Save Graph
7. Load Graph

# To-Do

1. Finish graphing mechanics
2. Register users
3. Delete graphs
4. Import/Export graphs from/to file (JSON format)
5. Google/Facebook Authentication
6. Explicit/Implicit Resizing
7. Pan & Zoom