# MEAN Stack

- Traditional Web development
  - Stack
    * HTTP: Apache/Nginx
    * Business logic: PHP/Servlet
    * Data: MySQL
  - (Almost) all code runs on the server
  - Browser is a passive HTML/CSS rendering engine
- After AJAX
  - Most code runs in the browser as JavaScript
    * Started as just fancy animation, but gradually more things get implemented as client-side JavaScript
    * Server is gradually transformed to be a "back-end service" that provides data-persistence and transaction support
  - Benefits
    * Better user experience
    * Less load on the server (leading to better server scalability)
  - Challenges
    * Increasing complexity in the JavaScript code
    * Impedance mismatch
      ‣ JavaScript on client, PHP/Java on server
      ‣ JSON for data transport, relation for data storage
  - Questions
    * Q: Can we use JavaScript for both client and server development?
      ‣ Node.js
    * Q: Can we use JSON as the data model for storage engine?
      ‣ MongoDB
    * Q: Any way to reduce complexity of client-side development?
      ‣ Angular, React, Vue, . . .
- Modern Full-Stack Web Development
  - Presentation: Angular/React/Vue, . . .
  - Business logic: Node.js + Node.js modules
  - Data: MongoDB