ARCHITECTING FOR ENTERPRISE WITH JAVASCRIPT
WHAT SHOULD I EXPECT?
BENEFITS OF USING JAVASCRIPT IN ENTERPRISE
SAME LANGUAGE ACROSS APPLICATIONS / PLATFORMS
<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>golazo-admin-4119103128-1yppat</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>10d</td>
</tr>
<tr>
<td>golazo-api-2684836542-dgL179</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>10d</td>
</tr>
<tr>
<td>golazo-consumer-3194053504-7m233</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-consumer-3194053504-9nk0n</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-consumer-3194053504-h22k0j</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-consumer-3194053504-1zz6i</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-consumer-3194053504-pluhh0</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-instagram-746611945-xoe1c</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>10d</td>
</tr>
<tr>
<td>golazo-opta-1577887311-omwlk</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>10d</td>
</tr>
<tr>
<td>golazo-phantom-1980129716-l5ub7</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>10d</td>
</tr>
<tr>
<td>golazo-realtime-1171874638-64nd9</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-realtime-1171874638-6zd0h</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-realtime-1171874638-8m0c0y</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-realtime-1171874638-njw5a0</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-realtime-1171874638-y2dmq</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-twitter-3346246250-e51f2</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-update-2812586172-d90k0</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>10d</td>
</tr>
<tr>
<td>golazo-varnish-2725234502-denn6</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
<tr>
<td>golazo-varnish-2725234502-infjy</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>3d</td>
</tr>
</tbody>
</table>
INCREASED VELOCITY
NODE.JS AT PAYPAL - JAVASCRIPT VS JAVA

• Built the same application in almost half the time
• Written with 33% fewer lines of code
• Constructed with 40% fewer files

https://www.paypal-engineering.com/2013/11/22/node-js-at-paypal
IMPROVED APPLICATION PERFORMANCE
Node.js Server

Request

Request

Requests

Requests

Event Loop

Delegate

POSIX Async Threads

Non-blocking IO

Thread Processing

Thread Waiting

https://strongloop.com/strongblog/node-js-is-faster-than-java
https://strongloop.com/strongblog/node-js-is-faster-than-java
What the heck is the event loop anyway?

@philip_roberts
LARGER ECOSYSTEM
http://www.modulecounts.com/
REGISTRY GROWTH

npm's growing significantly every day. Right now, there are about 11,165 new packages published per week. Of those, here are 32,768 of npm's most downloaded packages.

https://unpm.nodesource.com
LARGER TALENT POOL
I. Most Popular Technologies

<table>
<thead>
<tr>
<th>Language</th>
<th>2016</th>
<th>2015</th>
<th>2014</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>JavaScript</td>
<td>55.4%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQL (or SQL Server)</td>
<td>49.1%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java</td>
<td></td>
<td>36.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C#</td>
<td></td>
<td></td>
<td>30.9%</td>
<td></td>
</tr>
<tr>
<td>PHP</td>
<td></td>
<td></td>
<td>25.9%</td>
<td></td>
</tr>
<tr>
<td>Python</td>
<td></td>
<td></td>
<td>24.9%</td>
<td></td>
</tr>
<tr>
<td>C++</td>
<td></td>
<td></td>
<td>19.4%</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td>15.5%</td>
<td></td>
</tr>
<tr>
<td>Node.js</td>
<td></td>
<td></td>
<td>17.2%</td>
<td></td>
</tr>
<tr>
<td>AngularJS</td>
<td></td>
<td></td>
<td>17.9%</td>
<td></td>
</tr>
<tr>
<td>Ruby</td>
<td></td>
<td></td>
<td>8.9%</td>
<td></td>
</tr>
<tr>
<td>Objective-C</td>
<td></td>
<td></td>
<td>6.5%</td>
<td></td>
</tr>
</tbody>
</table>

BENEFITS OF JAVASCRIPT - RECAP

- Same Language Across Apps / Platforms
- Increased Velocity
- Improved Application Performance
- Larger Ecosystem
- Larger Talent Pool
JAVASCRPT
ALL THE THINGS
COMMON TRAPS AND PITFALLS
“VELOCITY IS THE MEASURE OF YOUR ABILITY TO ADAPT”

- Eric Elliot

https://ericelliottjs.com/product/tdd-es6-react/
MOVING TOO FAST
FRAGILE SYSTEMS
USING CLASSICAL INHERITANCE PATTERNS
WHY?
"FAVOR OBJECT COMPOSITION OVER CLASS INHERITANCE."

- The Gang of Four

Design Patterns: Elements of Reusable Object Oriented Software
CLASSICAL INHERITANCE IS OBSOLETE:

How to Think in Prototypal OO

Eric Elliott

https://www.youtube.com/watch?v=IKCCZTUx0sI
COMMON ISSUES FACED WITH CLASSICAL INHERITANCE

- Tight Coupling
- Gorilla/Banana Problem
- Duplication By Necessity

Fluent 2013 - Eric Elliot, "Classical Inheritance is Obsolete"
INCONSISTENT CODE STYLES AND ARCHITECTURAL PATTERNS
ENTROPY
EVERYTHING IS NOT A NAIL
<SoccerFeed TimeStamp="20160915T203134+0100">  
  <SoccerDocument Type="Result" uID="f842701">  
    <Competition uID="c98">  
      <Country>USA</Country>  
      <Name>US Major League Soccer</Name>  
      <Stat Type="season_id">2016</Stat>  
      <Stat Type="season_name">Season 2016/2017</Stat>  
      <Stat Type="symid">US_ML</Stat>  
      <Stat Type="matchday">7</Stat>  
    </Competition>  
    <MatchData>  
      <MatchInfo MatchType="Regular" Period="FullTime">  
        <Attendance>21144</Attendance>  
      </MatchInfo>  
    </MatchData>  
  </SoccerDocument>  
</SoccerFeed>
COMMON TRAPS AND PITFALLS - RECAP

• Moving Too Fast
• Fragile Systems
• Using Classical Inheritance (Deep)
• Inconsistent Code Styles / Architectural Patterns
• Entropy
• Everything Is Not A Nail
CODE QUALITY
KEEP IN MIND THAT I'M SELF-TaUGHT, SO MY CODE MAY BE A LITTLE MESSY.

LEMMME SEE— I'M SURE IT'S FINE.

...WOW.
THIS IS LIKE BEING IN A HOUSE BUILT BY A CHILD USING NOTHING BUT A HATCHET AND A PICTURE OF A HOUSE.

IT'S LIKE A SALAD RECIPE WRITTEN BY A CORPORATE LAWYER USING A PHONE AUTOCORRECT THAT ONLY KNEW EXCEL FORMULAS.

IT'S LIKE SOMEONE TOOK A TRANSCRIPT OF A COUPLE ARGUING AT IKEA AND MADE RANDOM EDITS UNTIL IT COMPILLED WITHOUT ERRORS.

OKAY, I'LL READ A STYLE GUIDE.
WHAT MAKES QUALITY CODE?
F.I.R.S.T. PRINCIPLE
F.I.R.S.T. PRINCIPAL

• Focused
• Independent
• Reusable
• Small
• Testable
WHETHER IT’S A CLIENT OR SERVER-SIDE COMPONENT, A NODE MODULE OR A PIECE OF VISUAL UI, COMPONENTS THAT ARE LARGE ARE INHERENTLY MORE COMPLEX TO MAINTAIN THAN THOSE THAT ARE SMALL

- Addy Osmani

https://addyosmani.com/first
DISCOVERABLE
“DISCOVERABILITY IS THE DEGREE TO WHICH YOU CAN FIGURE OUT HOW TO USE SOMETHING SIMPLY BY PLAYING WITH AND EXAMINING IT.”

- Erik Dietrich

http://www.daedtech.com
HOW CAN WE MAKE CODE DISCOVERABLE?
THAT IS AN EXCELLENT QUESTION
HOW TO MAKE CODE DISCOVERABLE

- Documentation
- Tests
- Typing
- Declarative Naming Patterns
- Cohesive File Structure
SETTING STANDARDS
Use npmjs registry and shrinkpack all projects
#835 opened 8 days ago by john-dodson

💡 ProTip! no:milestone will show everything without a milestone.
CODE QUALITY - RECAP

• Follow the F.I.R.S.T. Principal
• Code Should Be Discoverable
• It’s Important to Set and Enforce Standards
• PR’s Are a Great Way To Ensure Consistency
FINE TUNING YOUR ENVIRONMENT
“VELOCITY IS THE MEASURE OF YOUR ABILITY TO ADAPT”

- Eric Elliot

https://ericelliottjs.com/product/tdd-es6-react/
UNIT TESTING
WHAT DEFINES A UNIT?
“AN INDIVIDUAL THING OR PERSON REGARDED AS SINGLE AND COMPLETE BUT WHICH CAN ALSO FORM AN INDIVIDUAL COMPONENT OF A LARGER OR MORE COMPLEX WHOLE”

- The Dictionary
I DON'T GET IT
UNIT TESTING IS A SOFTWARE DEVELOPMENT PROCESS IN WHICH THE SMALLEST TESTABLE PARTS OF AN APPLICATION, CALLED UNITS, ARE INDIVIDUALLY AND INDEPENDENTLY SCRUTINIZED FOR PROPER OPERATION

- Tech Target

http://searchsoftwarequality.techtarget.com/definition/unit-testing
MEDIUMEST??
BLACK BOX TESTING
BLACK-BOX TESTING IS A METHOD OF SOFTWARE TESTING THAT EXAMINES THE FUNCTIONALITY OF AN APPLICATION WITHOUT PEERING INTO ITS INTERNAL STRUCTURES OR WORKINGS
“THIS METHOD CAN BE APPLIED TO VIRTUALLY EVERY LEVEL OF SOFTWARE TESTING...”

- Wikipedia

https://en.wikipedia.org/wiki/Black-box_testing
CLOSING THE FEEDBACK LOOP
MANAGING DEPENDENCIES
YOU BETTER BE PREPARED
DEPENDENCY MANAGEMENT STRATEGIES

- Shrink Pack
- Rollup
- Yarn
- Private Registries
- Docker
HANDLING ERRORS
PROCESS EVENTS TO HANDLE

- Uncaught Exception
- Unhandled Rejection
- SIGTERM
<table>
<thead>
<tr>
<th>Browser</th>
<th>Message</th>
<th>URL</th>
<th>lineNo</th>
<th>colNo</th>
<th>errorObj</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firefox 42</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Chrome 46</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Android Browser 4.4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Edge</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>IE 11</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>IE 10</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>IE 9, 8</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Safari 9</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>iOS 9</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

[http://blog.getsentry.com](http://blog.getsentry.com)
GETTING PREVENTATIVE
SECURING YOUR RUNTIME
NODE LONG TERM SUPPORT VERSIONS
N|SOLID FEATURES

• Package Vulnerability Monitoring
• 24 Hr Security Patch Integration
• Threshold and Process Monitoring
• Drop In Node Replacement
FINE TUNING YOUR ENVIRONMENT - RECAP

• Avoid Over Unit Testing
• Black Box Testing Can Help Solve Over Testing
• Closing the Feedback Loop Increases Velocity
• You Must Manage Your Dependencies
• You Must Handle Errors
• Securing Your Runtime Is Important
DEPLOYING WITH DOCKER
service:b01

STAGING
--tag prod

PROD

service:prod
DEPLOYING SANS DOCKER
DEPLOYING STATIC WEB APPS
AUTOMATION IS 🔑
WOW! WE’RE IDENTICAL!
ENGINEERS

DELIVERABLES
MONITORING & CUSTOM METRICS
APPLICATION PERFORMANCE MONITORING

- Requests Per Minute
- Throughput
- Time Spent in Execution
PROCESS MONITORING

• Memory Usage
• CPU Usage
• Network Usage
Gotta catch 'em all!
DEPLOYMENTS AND MONITORING - RECAP

• Deploying with Docker Provides Application Versioning
• Now.sh is a Great Solution for Node Deployments
• Deploying Static Web Apps Is Easy Too
• Automation is Key
• Stated Is Great For Custom Metrics
• Segment Is Great For Business Analytics
MANAGING RELIABILITY
THIS IS FINE.
SERVICE LEVEL OBJECTIVES
A TARGET VALUE FOR A SERVICE LEVEL THAT IS MEASURED BY A GIVEN METRIC

- Site Reliability Engineering

http://shop.oreilly.com/product/0636920041528.do
BENEFITS OF SLOs

• Easily measurable
• Quickly identify regressions
• Easy to align with KPIs
• Provides clear improvement paths
EXAMPLES OF SLOs

• 99% Service Uptime per Quarter
• Able to Serve 1000 RPM
• < 500ms Response Times
• 100k Concurrent Connections
• < 1s To First Usable Render
ERROR LIMITS
EXEMPLARY OF ERROR LIMITS

• 1% Service Downtime per Quarter
• < 900 RPM
• > 600ms Response Times
• < 90k Concurrent Connections
• > 1s To First Usable Render
MANAGING RISK - RECAP

• Managing Risk Allows You to Fine Tune Your Velocity
• Service Level Objectives Provide Goals for Systems
• Error Limits Are Used To Gauge Risk Introduced
CULTURE SHIFT
“YOU CAN EITHER EMBRACE CHANGE OR YOU CAN BRACE FOR CHANGE”

- Emily Webber

http://www.slideshare.net/turingfestival/emily-webber
FOSTER AN ENVIRONMENT OF LEARNING
POWER TO THE DEVELOPERS! 🥊
Simplifying the Future

https://www.youtube.com/watch?v=DGK6jjamzfY
EMPOWERING DEVELOPERS

- Purpose driven culture
- Whoever builds it, runs it
- From project to product
START SMALL
WHICH ONE?
CULTURE SHIFT - RECAP

• You Must Embrace Change
• Foster An Environment of Learning
• Give Power to Your Developers
• Start Small
ARCHITECTING FOR ENTERPRISE WITH JAVASCRIPT

: @kurtiskemple

: @kurtiskemple

: @kkemple