Final Presentation

Team D

Bryan
Charles
Donovan
Jonathan
Katie
Kia
Tyler

github.com/Cap7pdx

The Product

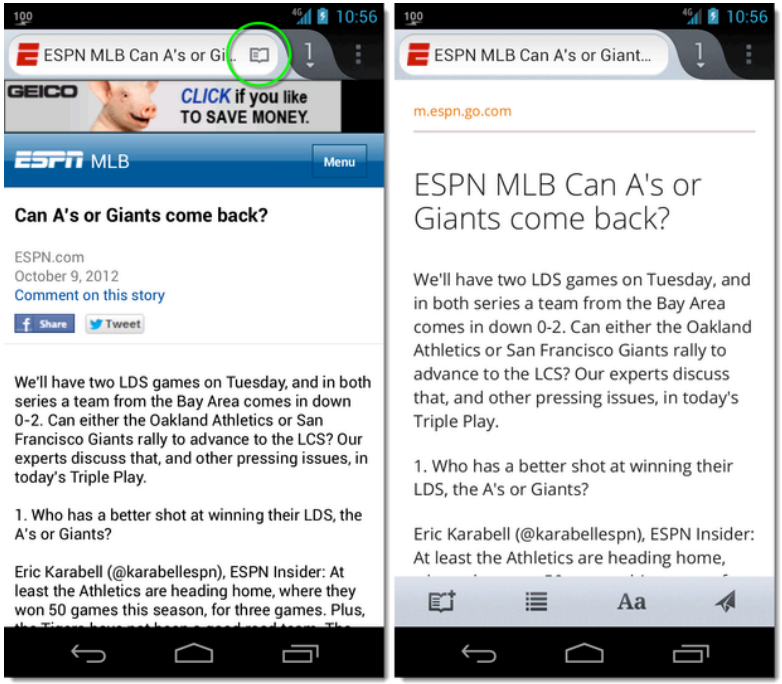
What does our product do?

Runs on a phone! maybe.

Proof of concept to improve content detection of a website

Determine what content is relevant to a site, while retaining original context

Only send that relevant data to the device



The Product

Why is it needed?

Optimize how a website is rendered, agnostic of device

Low bandwidth and/or expensive bandwidth

Non-mobile friendly sites

Accessibility

Who will use it?

Users with limited or expensive bandwidth

Anyone using Mozilla's Firefox OS or other devices running Firefox

Assumptions and Constraints

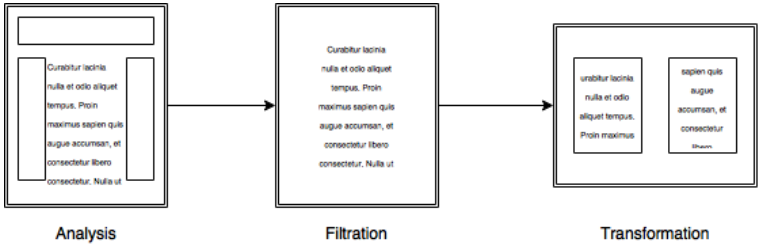
Assumptions

- Websites with similar content have similar structure
- Relevant content can be identified with a (relatively) naive algorithm
- A small-ish set of data can relevantly inform how the algorithm operates on the whole internet
- Development could proceed concurrently with research

Constraints

- (Originally) Must run on Firefox OS phone
- (Originally) Must use Firefox OS browser API
- (Originally) Must run in Firefox OS (written in JavaScript)
- Unable to perform pre-emptive filtration without access to internals of browser

Features



What features does your product have?

Content Analysis: Detect and traverse the structure of a web page

Print the nested structure of a page to understand the contents in it

Access the document object model to dynamically update the content, structure and style of the document

What features did you originally plan for it to have?

Content Filtration: Remove unwanted/extra contents without affecting user's experience

Content Transformation: One that would be based on the client and the device. When removing the contents in **Filtration**, the

Deliverables

Working prototype written in JavaScript

Switched from a Firefox OS app to a NodeJS one.

Browser App to Command Line program outputs HTML

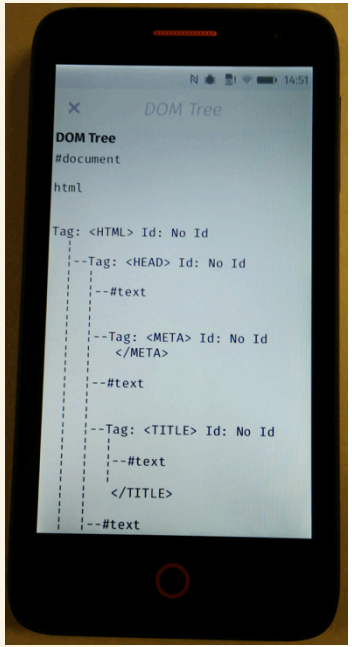
https://github.com/Cap7pdx/phoenix-node

```
dfinch@combusken $ node alt.js
<HTML><HEAD><h1>DOM Tree</h1><TITLE>DOM Tree</TITLE></HEAD><BODY>

<CODE>

#documenthtml


<HTML>
  |
  |--<HEAD>
  |   |
  |   |--<META>
  |   |   <META>
  |   |
  |   |--<META>
  |   |   <META>
  |   |
  |   |--<META>
  |   |   <META>
  |   |
  |   |--<META>
  |   |   <META>
  |   |
  |   |--<TITLE>
  |   |   <TITLE>
  |   |
  |   |--<SCRIPT>
  |   |   <SCRIPT>
  |   |
  |   |--<STYLE>
  |   |   <STYLE>
  |   |
  |   |--<STYLE>
  |   |   <STYLE>
  |   |
  |   |--<SCRIPT>
  |   |   <SCRIPT>
  |   |
  |   |--<LINK>
  |   |   <LINK>
  |   <HEAD>
  |
  |--<BODY>
  |   |
  |   |--<SCRIPT>
  |   |   <SCRIPT>
  |   |
  |   |--<DIV> Id: mngb
  |   |   |
  |   |   |--<DIV> Id: gbar
```



Deliverables

Research paper detailing our findings

"Methods for Web Content Analysis and Context Detection"

Content Extraction (e.g. scraping) is done by mostly proprietary systems.

Readability

Prior research:

"Boilerplate detection using shallow text features" C. Kohlschütter, 2010

-> DOM Distiller -> Chrome Distill Page & mobile Chrome "Mobile-Friendly mode"

"Evaluating Web Content Extraction Algorithms" T. Kovacic, 2012

Reviewed 17 simple & machine learning modes for extracting text content

Process and Schedule

- Scrum
  - 2-Week Sprints
  - Sprint Transition
    - Backlog Grooming
    - Sprint Retrospect
    - Sprint Planning
- Tools
  - Pivotal Tracker
  - GitHub
  - Toggl
  - Slack
  - Google Drive

Process and Schedule

Sprint 01	Sprint 02	Sprint 03	Sprint 04
(11/16/15 - 11/29/15)	(11/30/15 - 12/13/15)	(12/14/15 - 12/27/15)	(12/28/15 - 1/10/16)
Project Slides	Research: Web Structs.	Research: UI	Product Demo
“Phoenix Browser”	SQA: Benchmarks	Research: Pattern Detection	
	“Feather”		

Process and Schedule

Sprint 01	Sprint 02	Sprint 03	Sprint 04	Sprint 05
(1/04/16 - 1/18/16)	(1/19/16 - 2/01/16)	(2/02/16 - 2/15/16)	(2/16/16 - 2/29/16)	(3/01/16 - 3/07/16)
Background Research: DOM	Research: Web Structures	Research: Reader Modes	Compile Research Data	Deliver Research Paper
Background Research: Browser API	Research: Pattern Detection	Research: Various Algorithms	Write Paper	Update Paper Based on Feedback
“Feather” 1.0	“Feather” 1.5	Research: Cluster Analysis	Write Blog Post for Firefox Website	

Team Roles

Person	Planned Role	Actual Role
Bryan	Dev/SQA	Research
Charles	SQA/Unit Testing	Research
Donovan	Dev/SQA	Research
Jonathan	Team Lead/Dev/SQA	Team Lead/Research
Katie	Dev Team	Research
Kia	SQA/Unit Testing	Dev Team
Tyler	Dev Team	Research

Problems and Contingencies

Event	Mitigation
Mozilla stopped producing Firefox OS phones	Firefox OS codebase still exists and will continue to be developed
Loss of a team member	Other team members absorbed duties and responsibilities
Cross-site scripting error when implementing browser	Found another way to implement the feature
Project was more research-focused than initially thought	Focused on research in Capstone II; designated a research team

Lessons Learned - Addressing our assumptions

We assumed development could proceed concurrently with research.

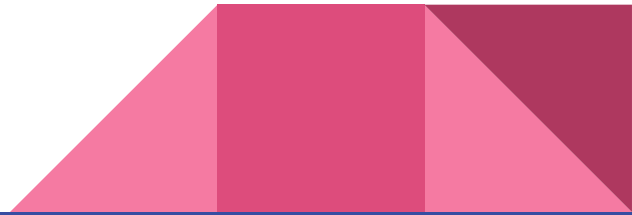
We should have spent more time understanding the domain and the problems before building.

Similar website content does not imply similar structure.

Counterexample: front pages of news websites as opposed to websites like Pinterest or Instagram.

Training a more complicated classification model requires a ton of data.

It is not trivial to identify relevant content.



Lessons Learned - Overall

Clarifying requirements is important.

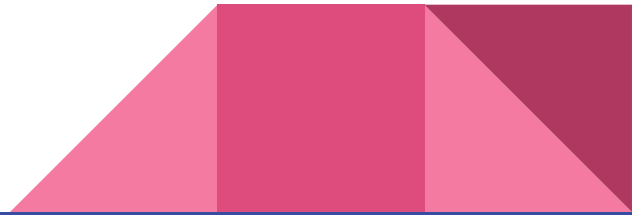
Unclear requirements and lacking domain knowledge made our sprint goals vague.

For this reason, our sprint goals sometimes lasted multiple sprints.

A project's scope, importance, or relevance can change.

We wasted man hours from previous, now irrelevant, sprint goals.

Our project became more abstract, so we had to be more flexible as developers and researchers.



Lessons Learned - Overall

A machine learning approach is more suitable for classifying website structures.

Our team communication improved over the term.

We all made adjustments to what and how we communicated over Slack to make the most progress.

It was hard to find a time all of us could meet with our sponsor, so our weekly memos helped greatly.

Our team organization also improved.

Using project management, time tracking, and other tools helped us assign work effectively.

Having this structure encouraged members to take on more responsibilities naturally.

Thank you!