

CUNY Academic Commons - Design/UX #10439

Design/UX # 13998 (New): Homepage Redesign

Create Style Guide for Commons

2018-10-04 05:39 PM - Sonja Leix

Status:	New	Start date:	2018-10-04
Priority name:	Normal	Due date:	
Assignee:	Sara Cannon	% Done:	0%
Category name:	Design	Estimated time:	0.00 hour
Target version:	1.20.0		

Description

Create a style guide for CUNY Academic Commons to compile all the site's design elements like typography, link, and button styles, color scheme, consistent padding/margin, etc.

This will help us design more consistently in the future, it will also help the developers to have a reference for implementation. It will be a base to achieve more design consistency across the site and aid as a style library for a potential future redesign.

It will be a living guide and after initial design and internal sign off we should consider creating an HTML style guide. It can live on Github and can be updated over time.

History

#1 - 2018-10-04 05:55 PM - Sonja Leix

I would approach it by first taking inventory of all existing styles on the site (I assume there are a lot of unnecessary duplicates or variations of styles), then make decisions together which ones to drop/replace to simplify the style guide to achieve more consistent styles. We want to only include styles we need, e.g. one style for each of these: primary button large, primary button small, secondary button large, secondary button small, link style in text, nav styles (default, hover, active), etc.

#2 - 2018-10-04 05:57 PM - Sonja Leix

Matt,

You mentioned that Chris might have started on a style guide or has done some work on buttons in the past? Is that something you could dig up and share? If not, I'm happy to start from scratch.

#3 - 2018-10-05 02:16 PM - Boone Gorges

Reposting Chris's comment from [#9801](#):

have to disagree about having Sonja spending time doing a style guide or inventory of styles of the current site. It's a mess and something that we would basically throw out when we redesign. But I do think there are things we could do to document functionality and needs.

Along those lines I have this site I made 3 or 4 years ago that is a start at documenting the current interface elements:

<https://redesign2015.commonsgc.cuny.edu/>

Sonja, I sent you an invite to the site. Let em know if you don't get it.

Here is the document on buttons that you were referring to Matt: <http://www.teachingmultimedia.com/commons/buttons/buttons.html>

If you read it you'll see that there are a number of different styles which are sometimes confusingly used. [Some of these issues have actually been remediated.]

Instead of documenting the styles that we have, I would instead advocate that time be spent documenting functionality and types of interface elements. If we do that it could be a valuable tool for the redesign.

So at a higher level than buttons, but maybe how much do we have an list of items with title, thumb, date, description, or when do we use tabs, etc.

For the redesign we can then focus on whether these are the right interfaces, making a style guide etc.

I would also advocate against what I'm doing here and move the conversation on style guides and redesign to another thread either here in red mine or on the Commons.

#4 - 2018-10-05 02:19 PM - Boone Gorges

To respond to Chris's feedback above: I take your point, but would like to push back against it. For five+ years we have put off incremental visual and design improvements because it felt like a waste to put effort into something in advance of a major redesign. As a result, there are small or medium things we might have addressed years ago that would, in hindsight, have been worth handling, but which have gone undone because of our approach. Let's not continue to do this. If Sonja were to spend a few hours on a styleguide - first, doing a high-level review of interface elements and design, as recommended by Chris, but then also documenting (within reason) existing styles - it's likely that it'd lead directly to us making some meaningful, small changes that would serve us well in advance of a full redesign.

This is predicated on the assumption that Sonja wouldn't be spending many dozens of hours on this project. If that's the case, then I agree that we should limit the scope.

#5 - 2018-10-05 05:56 PM - Chris Stein

Thanks for moving this over Boone. I hear what you're saying about the continually delayed changes and its a valid point. If Sonja's time is limited and focused on making the style guide rather than documenting all of the various styles in existence then I'm on board. Those two links represent a lot of hours looking over the site and comparing things. It can be a time sink. That's my main concern. having a styleguide would be great now and in the future.

Sonja I'm also open to your opinion on how best you can spend your time towards improving the existing interface while helping to set up the major redesign. I can also help doing some of the documentation or other work to help reduce busywork hours for you.

Also there are different versions of style guides, static/dynamic etc., can you tell us a bit more about the form you're thinking about for the guide?

#6 - 2018-10-08 06:01 PM - Sonja Leix

Thanks Boone for your input! I agree with the approach to not push this off yet again, but use this project to prepare for a major redesign.

Chris Stein wrote:

Thanks for moving this over Boone. I hear what you're saying about the continually delayed changes and its a valid point. If Sonja's time is limited and focused on making the style guide rather than documenting all of the various styles in existence then I'm on board. Those two links represent a lot of hours looking over the site and comparing things. It can be a time sink. That's my main concern. having a styleguide would be great now and in the future.

Sonja I'm also open to your opinion on how best you can spend your time towards improving the existing interface while helping to set up the major redesign. I can also help doing some of the documentation or other work to help reduce busywork hours for you.

Also there are different versions of style guides, static/dynamic etc., can you tell us a bit more about the form you're thinking about for the guide?

Thanks for sharing those resources and previous attempts. We can certainly use some of these resources to inform this task and give us a starting point.

I agree that given my somewhat limited time on the project, I wouldn't do a full style audit of the website, as I agree a lot of it would likely be wasted leg work. I would rather document the primary design elements (like buttons, typography styles, form elements, etc.) as well as interaction patterns (incl. things like error messages, etc.). From there I would start creating a living style guide (with the help of Boone).

After the first iteration of the living style guide is done, we can focus on high-traffic sections of the site and identify any missing styles. This will help us make it more and more comprehensive - along this journey we'll likely make a lot of decisions to simplify current styles. We can then start rolling out styles into the site over the next release(s).

This will also be the perfect opportunity to craft a comprehensive design system which will lay the foundation of the redesign. We currently have an extensive list of individual or contextual styles for elements. Which is, understandably so, the current reality due to iterating on the site over the past 5+ years.

How does that sound for scope? Happy to discuss and refine and am always open for suggestions.

#7 - 2018-10-08 11:12 PM - Matt Gold

This all sounds great to me. I appreciate the conscientiousness you are all showing, and Chris's concern that Sonja's time be spent wisely. I do agree that we need to start to move this all forward, and it sounds like Sonja will be able to recognize and step back from potential time-sinks as they approach. So, I would encourage us to move forward on this. Thank you all!

#8 - 2018-10-09 12:19 PM - Sonja Leix

Matt Gold wrote:

This all sounds great to me. I appreciate the conscientiousness you are all showing, and Chris's concern that Sonja's time be spent wisely. I do agree that we need to start to move this all forward, and it sounds like Sonja will be able to recognize and step back from potential time-sinks as they approach. So, I would encourage us to move forward on this. Thank you all!

Thanks Matt, I'll start working on this as soon as the 1.14 release issues no longer need my attention.

#9 - 2018-11-13 09:07 AM - Sonja Leix

For this project I'm looking to create a living style guide (incl. pattern library) – it will be driven by design, but live in code. This way we can all access the same most current version of the style guide. And ideally it's setup in a way that is directly connected to inform the styles of the website – so we only have to make an update to a style in one place. Or vice versa, the style guide is informed by the styles of the website and rendered from there to be up-to-date.

I've done some research into living style guide solutions. But first let me start by sharing a few examples of living style guides out there that I like:

Lonely Planet:<https://rizzo.lonelyplanet.com/styleguide/design-elements/colours>

Buffer:<https://buffer.com/style-guide>

CenturyLink:<https://assets.ctl.io/cyclops/1.8.6/>

Mailchimp:<http://ux.mailchimp.com/patterns/forms> (not sure if we need actual html code snippets in the style guide)

Chameleon:<https://pusher.github.io/chameleon/components/buttons/>

U.S. Design System:<https://designsystem.digital.gov/components/>

We can start with basic styles (like consistent buttons) and work our way towards a more complete style guide. This will help us work towards a full or partial redesign potentially in 2019.

Now we need to look at the technical aspect of living style guides and find a solution that works for us. I'd love to hear Boone's feedback on which solution would make the most sense and is the easiest to integrate and update. I did some research already, but would need to know which direction to go to dig a little deeper for the best solution:

Overview of Pattern Library Generators:<https://github.com/davidhund/styleguide-generators> (not sure which direction, PHP based, parsing CSS source, Gulp/Grunt, etc.)

More styleguide tools:<http://styleguides.io/tools>

Chameleon:<https://github.com/pusher/chameleon>

Pattern Lab:<https://patternlab.io/> (might not be exactly what we are looking for, but it's a holistic approach we potentially want to look at for the full redesign)

I worked with KSS on Node before:<http://kss-node.github.io/kss-node/>

First two links are overview of a variety of tools. Boone, if you could narrow it down for direction of tool or language, I can continue to dig.

#10 - 2018-11-13 03:05 PM - Boone Gorges

Thanks for sharing the initial research, Sonja! I've added Ray as a watcher, as he may have opinions regarding implementation.

As a general principle, I think we should steer clear of tools that are too closely tied to a specific templating language. A number of the tools try to enforce atomic design by asking you to build a library of reusable elements, and then to include them when building molecular or template-level items. This approach won't work well for us, for a couple reasons. One is that we use WordPress, which uses PHP-based template, rather than Moustache or Handlebars or a true templating language. (We could build middleware, but it's lots of work.) Two is that our markup comes from many places other than theme templates, such as plugins and mu-plugins. Finally, I can envision a future where we build things that are technically different from what the Commons currently is (say, something non-WordPress-based), but where we still want to use the style guide toolset.

I like the idea of building a styleguide out of our CSS, using KSS. kss-node is one tool of this type, <http://styleguide.sc5.io/> is another, and there are probably others. With this kind of approach, we could keep our existing infrastructure (ie not rewrite template files, and our changes to CSS would be mostly for consolidation at first), and add styleguide integration through inline comments + some build steps (grunt, php, whatever). This feels to me like something that we can do in stages, within our existing theme/plugin setup.

#11 - 2018-11-22 11:10 AM - Sonja Leix

Matt and team,

which part of the Commons should I first focus on to review what styles exist and to start creating a style guide? Please include any sub sections that might be relevant and you wanted me to also include in this.

To inform this questions the consideration should be most used or having the widest range of UI variation. Please let me know and if you can share a link so I'm sure to find the right thing.

Thanks.

#12 - 2018-11-23 02:37 PM - Matt Gold

Thanks, Sonja. I see a few possibilities here:

1. The profile/portfolio section -- e.g. what you see when you go to [https://commons.gc.cuny.edu/members/\[username\]](https://commons.gc.cuny.edu/members/[username]) and then also click on "Commons Profile." The rationale for starting here is that we have two divergent styles (public portfolio, which is its own design that the Commons team created a few years ago in an effort to help people put together online CVs) and the Commons profile, which inherits basic buddypress styling from the main site and which connects more to the network. So, what unites this area of the site is the Users own profile; but then there are two parts, one more public facing and one that shows internal connections with the Commons (e.g group memberships, sites, etc). FYI, when we were creating the Commons profile section, we had lots of discussions about what it meant to be creating a part of the site focused on the individual member (the portfolio) and a part of the site that focused on connections (which is what the Commons is supposed to be about). in the end, we split the difference and made both interfaces available through tabs.

2. The group interface. the rationale for starting here is that it is heavily used and also in need of a redesign. Work on this part of the site could also affect other areas of the site, such as group and blog directories or the Commons user's Commons profiles, as described above.

Those are probably the two places I'd start, though I'd love to hear what others think.
Commons Profile

#13 - 2018-11-26 01:58 PM - Boone Gorges

Starting with single groups seem like the best approach to me. As Matt notes, they're heavily used. They also contain much of the design language used throughout the site. Specifically, they have:

- single-item navigation (left-hand nav)
- secondary nav (like "All Docs" when clicking the Docs item)
- BP item lists (like under Members)
- miscellaneous other kinds of item lists that come from other plugins (Docs, Forums, Files)

#14 - 2018-11-27 02:21 AM - Sonja Leix

Thanks for sharing your feedback Matt and Boone.

Reviewing the suggested sections, I tend to lean towards Groups or more specifically starting with single groups. My reasoning is, that there is less of a focus on BuddyPress UI (though it's woven in with the members section). Groups include a lot of different styles and UI components that will help kick-start this project. I would start with the main single group page, and then start to dive deeper into the sub sections to identify unique styles that need to be defined. Once that's done, I can then focus on the groups archive page.

If everyone is ok with this direction, I'd start there.

#15 - 2018-11-27 09:53 AM - Chris Stein

I agree Groups is a good place to start.

While you're looking at groups, and the Commons in general a few things you might keep in mind:

There are a few main types of interfaces that are common across the site but not all the same

- Lists of things (people, groups, sites, activity). These tend to have similar elements (title, image, description, time) but not always similar style
- Forms
- Tabs
- Buttons and Links

As you move forward after that another thing...

There are some specialized interfaces that were designed more to fit their function than an overall style (this is off the top of my head and not exhaustive):

- Editing a Doc
- Social Paper
- Portfolio
- Inviting people to the Commons

I can also give you links to some of the research that was done a while ago like the Button Styles.

Perhaps if you like and have the time we can schedule a time to talk and I can cover some of the discussion Matt referenced on Profile/Portfolio and any other back history that you're interested in.

#16 - 2018-11-27 01:45 PM - Matt Gold

so -- groups it is! thank you, Sonja, and thank you, Chris

#17 - 2018-11-27 10:53 PM - Sonja Leix

Chris,

Thanks for the additional info and yes, would love to get on a call with you to get more insight. Do you have some time next week (Wed or Thur maybe)?

#18 - 2018-12-14 11:20 AM - Sonja Leix

- *File Commons-styleguide_121218.pdf added*

Hi team,

I've put together the first draft of a style guide focusing on Individual Groups.

The attached PDF includes:

- **Atomic Design Principles** – A look at what desired design system we're trying to achieve longer term
- **Styles inventory** – What styles are currently present on individual groups pages
- **Style guide** – A start of some general styles (*will be extended on in the next step*)

Understanding the longer-term design system we're trying to get to will help us understand how we think about elements and styles on the page.

The inventory will help us consolidate styles and make decisions about visual hierarchy.

The decisions we make by looking at the inventory will then be added to the style guide. From there we can start implementing styles for groups.

The goal is to create more consistency in design (and on the technical side HTML classes to style i.e. primary buttons the same across the site) and at the same time, make it easier to update styles. This will give us a big leg up during a possible redesign project.

#19 - 2018-12-14 03:19 PM - Boone Gorges

This is excellent, Sonja!

What are next steps? Perhaps we can start by choosing one or two items to remediate - say, consolidating the various greys - and use that as a test case for integrating the living styleguide toolchain?

#20 - 2018-12-18 11:52 AM - Sonja Leix

Boone,
Yes, I'd like to start with a few items. The greys would be a good starting point, so would be the buttons. I can put together a unified look for primary, secondary, and maybe another type of button.

#21 - 2019-04-23 03:54 PM - Boone Gorges

- Target version changed from 1.15 to 1.16

#22 - 2019-05-17 06:23 PM - Sonja Leix

Matt,
it would be good to discuss in what capacity you want me to focus on this since the redesign bid is also something that will be going out at some point.

#23 - 2019-11-14 04:21 PM - Boone Gorges

- Target version changed from 1.16 to 1.17.0

This is an ongoing task.

#24 - 2020-02-28 06:23 PM - Sonja Leix

Hi team,
As I discussed with Matt, once I've wrapped up the designs for the 1.17 release, I'll get back to this task and the greater redesign of the Commons — so excited to get this started!!!

I have a few questions in preparation of that, which I'd like to talk to the team about during our Tuesday meeting. I'd also like to share a few thoughts about overall strategy and approach with everyone. Here are the topics I'd like to discuss:

- Component based approach and overall design strategy
- Is Redmine the right tool for this project or would we use a repo like Github for this project (thinking about keeping track of relationships between components and overall progress etc.)
- Does the dev team have any preferred or true and tested living style guide tools they'd like to use?
- Does my Commons user profile have full access to all relevant areas of the website we're redesigning? What sections might I be missing?
- Timeline and communication

#25 - 2020-07-14 01:05 PM - Boone Gorges

- Target version changed from 1.17.0 to 1.18.0

#26 - 2020-12-08 11:31 AM - Boone Gorges

- Target version changed from 1.18.0 to 1.19.0

#27 - 2021-01-14 01:17 PM - Sonja Leix

Hi team,
I've researched some possible solutions for our living style guide as part of our redesign project.
It was important to look at open source solutions to be inline with our mission, as well as for a flexible and scalable solution.

Pattern Lab

Pattern Lab is a frontend workshop environment that helps you build, view, test, and showcase your design system's UI components. Built on Atomic Design principles.

<https://patternlab.io/>

Demos: <https://patternlab.io/demos/>, <https://altinn.github.io/designsystem-styleguide/tekst/artikkelheader.html>

A few notable features:

- Open Source
- Nested patterns (will be great for our component based approach)
- Pattern documentation (define and describe UI patterns for cross team understanding)
- Responsive views (to look at components across various screensizes)
- Tool and language agnostic, lives on Github and has support and resources
- "Pattern lineage" (see where components are used)
- Includes a pattern starter kit and is flexible and extensible
- Interactive components

Catalog

Open Source

Catalog combines design documentation with real, live components in one single place, making collaboration between designers and developers seamless.

<https://www.catalog.style/>(<https://www.catalog.style/>)

Documentation: <https://docs.catalog.style/>

Demos: <https://brand.interactivethings.com/#/>, <https://www.interactivethings.com/styleguide/#/>

A few notable features:

- Open Source
- Easy to maintain
- beautiful presentation
- Direct integration with React applications (not sure if that will work for us)
- Built on modular system, flexible and customizable
- Add Design guidelines, document and specify styles
- Interactive components
- Responsive views
- Lives on Github, has documentation and support

Emulsify

Emulsify is an open-source tool for creating design systems with reusable components and clear guidelines for teams.

<https://www.emulsify.info/>

Documentation: <https://docs.emulsify.info/>

I can't tell for certain, but this solution might be for Drupal only

A few other tools I've found, but that don't seem to be open source:

Style Guide: <https://hugeinc.github.io/styleguide/>

Fabricator: <https://fbrctr.github.io/>

Style Guide Guide boilerplate: <https://bradfrost.github.io/style-guide-guide/>

I'd like to ask Boone and the dev team to look at these solutions to see which ones are viable from a technical perspective. I imagine it's most important to be able to seamlessly integrate with our environment and the dev workflow. If none of those work for us, I'm happy to do more research.

#28 - 2021-01-14 11:55 PM - Chris Stein

Hi Sonja, these all look great, thanks for compiling them. I'll leave it to others who are doing more direct development for comments. The only other open-source tool I've come across is Storybook which seems to be both a tool for developing/testing UI Components and with Addons can output documentation.

<https://storybook.js.org/>

<https://medium.com/storybookjs/how-design-systems-use-storybook-2ed735ad07a9>

#29 - 2021-01-15 01:36 PM - Sonja Leix

Chris Stein wrote:

Hi Sonja, these all look great, thanks for compiling them. I'll leave it to others who are doing more direct development for comments. The only other open-source tool I've come across is Storybook which seems to be both a tool for developing/testing UI Components and with Addons can output documentation.

<https://storybook.js.org/>

<https://medium.com/storybookjs/how-design-systems-use-storybook-2ed735ad07a9>

Thanks Chris, that looks like a viable option for us too from my perspective. Would love for the dev team to look at this one too.

#30 - 2021-01-20 06:52 PM - Boone Gorges

Thanks for doing the initial research.

Some constraints that are guiding my thinking:

1. We are working with an existing codebase, mostly WordPress theme templates but also things like CSS, miscellaneous PHP files (global footer), etc.
2. We commonly build UI using at least two different technologies. First, and most commonly, we use WordPress theme templates. Second, we have an increasing number of JS-driven interfaces. These latter interfaces are split between those built in jQuery/vanilla JS, and those built in Vue.js.

These constraints are important as we consider a path forward. Most of the tools suggested here are "design systems" with very broad goals. They all allow you to build a beautiful style guide. Most have ways to make this style guide be "living", in the sense that your user-facing application (in our case, the WordPress theme) is built out of the very same pieces as the style guide. To accomplish the "living" aspect, the tools offer different ways of integrating into design and development workflows, in ways that are more or less flexible.

Because of our constraints, flexibility is extremely important. The same modesty that is guiding our redesign - "one piece at a time" - must guide any major modifications to our development technologies and workflow. As such, I think we have to rule out any systems that are all-or-nothing, systems that would require a massive redesign of our technical workflows just to be able to begin integration.

Of the options that I've looked at here, Pattern Lab seems to be perhaps the safest bet. It seems to be the most popular and well-established, giving me more hope that it'll continue to be around for a long time. It also seems to be the most flexible. Emulsify, for example, seems pretty cool, but it

seems to require a specific design-dev flow. Pattern Lab, on the other hand, seems to be somewhat agnostic in terms of how you build your application, since it mainly spits out Mustache or Twig templates and lets you figure out (with the help of third-party tools) how to integrate this into your app.

Our team will need to have a couple of parallel discussions about strategy and vision, both short- and long-term, for the styleguide and the future of design-driven development on the Commons project. Here are some initial directions for those discussions.

Technical and procedural workflow for style guide construction and hosting

Pattern Lab works by defining a set of templates and other primitives ("atoms", "molecules", etc) in the /source/ directory of the pattern-labs application. These primitives can take a number of forms: scss files, Handlebars templates, JSON config files, etc. An npm script is used to build the public-facing styleguide.

Our team would need a workflow describing how design ideas get from designer into the pattern library. For example, we might have a setup like this:

- a. We create a new Git repository for our /source/ files.
- b. Members of the team have a local version of Pattern Lab. This includes Sonja (as designer) and me (as release manager) and anyone else who needs it.
- c. Sonja makes suggested changes to our pattern library. She could either make the changes directly to the file and commit them to the repo. This is straightforward for some changes, like hex codes. In other cases, she may deliver the requested changes to the development team, who would then change the pattern library and commit the changes.
- d. The PL build scripts create a static HTML site. At first, we'll probably build and deploy this site manually. But we could make it automatic in the future - say, a GitHub action that publishes to GitHub pages on every commit to the pattern library repo.

Integration into the Commons application

We are working with massive codebase, built on legacy technologies like WP templates. We cannot swap everything out right now. But I can imagine a phasing-in process, where the styleguide becomes more "living" with each step. For example:

- a. At first, there is no integration with the Commons codebase. The pattern library is simply a reference for the development team, who is then responsible for manually applying the patterns to the WordPress application.
- b. Certain static assets, like image files (think: logos, AJAX spinners, other glyphs) may be a good first candidate for integration. We can either: (i) load these images directly from the pattern library, perhaps by including the pattern library as a submodule and deploying it along with the main site, or (ii) add some sort of build step that copies files as needed, pre-release.
- c. Sass integration may be a good next step. Variables for things like colors and typography could be defined in the pattern library, and we could pull them into the WordPress theme as part of a build step. This would involve moving to Sass for the Commons theme, and also introducing build/watch tools.
- d. Markup templates - more significant 'molecules', 'organisms', etc - are more complex. For example, it would be good to have a centrally defined pattern for each type of button. PL would generate these as either Handlebars or Twig templates. Say we use Handlebars. These templates would then be usable in jQuery templates by loading the Handlebars library, and it might be possible to import at least the smaller of them for use in Vue.js. For PHP, we would need a build step that turns them into PHP (see eg <https://github.com/zordius/lightncandy>). (Something similar could happen with Twig, but we'd need middleware for templates on the JS instead of the PHP side.) In our WordPress templates, then, we would load these built templates by passing variables to `get_template_part()` (or some other convention that we would have to invent).

These are all just initial thoughts on technical integration. As noted above, PL is pretty agnostic about integration, so we would have flexibility to experiment. It's worth noting that these changes will make our development and deployment process more complex by introducing build steps, but perhaps this just brings us into the modern age of web applications.

Maybe we could take a bit of time to discuss next steps during our next meeting.

#31 - 2021-01-25 04:42 PM - Sonja Leix

Thanks Boone for this detailed response! All the steps and considerations you've shared (at least the technical ones I could follow ;) make a ton of sense to me.

I want to add one more option here which could possibly be part of the interim step or longer term goal for implementation of styles in WordPress templates. Couldn't we also define clear css selectors that help create a consistent design. For example padding and margins could be globally defined with a few selectors, etc.

Looking forward to discussing this more tomorrow.

#32 - 2021-01-26 03:13 PM - Raymond Hoh

About Pattern Lab, just did some brief research and there are a few WordPress integrations available on Github.

We can probably take some inspiration from those to get set up. Here are some implementations:

- <https://github.com/forumone/gesso-wp>
- <https://github.com/macder/derulski.com>

Along with Pattern Lab, both of these implementations uses [Timber](#), which is a WordPress middleware development library that utilizes [Twig](#) as the templating engine. Tying ourselves to another library like Timber might hinder us in the future, but it looks like a lot of people use Timber.

We are working with an existing codebase, mostly WordPress theme templates but also things like CSS, miscellaneous PHP files (global footer),

etc.

If we move to a template engine, things like the global footer wouldn't work in the context of other themes, so we'd need to keep a copy of the global footer PHP file for compatibility reasons.

#33 - 2021-01-26 04:54 PM - Boone Gorges

Thanks, Ray. I spent some time looking at Twig + Timber, but I hadn't seen those plugins. I'll take a deeper look. Twig seems like it's widely-enough used that I feel OK thinking about a future where we introduce it incrementally. I think the added value of Timber is that it provides standardized data objects for things like posts and users, which may have limited utility in our highly customized environment (particularly as regards BP objects). In any case, as I mentioned above and in the call, the Pattern Lab approach is somewhat agnostic as far as whether and to what extent we integrate, so these are questions we don't have to decide today - it's just reassuring to know that there are options down the road.

I'll take time in the next week for further review and thoughts about next steps.

#34 - 2021-02-01 02:01 PM - Colin McDonald

Hello, just thought I'd bump this before our call tomorrow. I know it's a busy time with the semester beginning, and perhaps this is largely confined to the core dev team right now, but just in case there's anything for Sonja and others on the call to help evaluate.

#35 - 2021-02-01 02:06 PM - Boone Gorges

Sorry, I'm continuing to do research on it, but I don't have anything to share yet. I will continue working on it, though I can't promise to have conclusions by tomorrow's dev call.

#36 - 2021-02-16 06:13 PM - Colin McDonald

- Parent task set to #13998

#37 - 2021-03-10 03:20 PM - Boone Gorges

Hi all - An update from the technical end.

Using Pattern Lab <https://patternlab.io/>, I've put together the framework for a styleguide. Here is the repo: <https://github.com/cuny-academic-commons/commons-style-guide/>. Setup instructions are in the readme file.

I've written a GitHub Actions workflow to build and deploy the styleguide to GitHub Pages on every push, so that you can see the styleguide in action at <https://cuny-academic-commons.github.io/commons-style-guide/?p=all>

I started with some of the basic patterns that Pattern Lab ships with, and used them to learn how to use PL. A few, I customized to match what we actually have on the Commons. See eg Atoms > Tokens > Brand Colors. A bunch of them I just left in place with the default values - I'll remove them as we get started on the actual work.

I also built out our Directory Page, to get a sense of how entities might be organized - again, primarily as an exercise in learning how to use PL. See Pages > Directory. It's not perfect, but it pretty closely matches the style and functionality of our directory pages, but gutted and rebuilt for PL. How this works under the hood is too complex for this space, but you can get a sense of the potential for modularity and atomic design by noticing how the Page is built from multiple organisms (Organisms > Directory), which in turn are made out of molecules (Molecules > Directory). The inheritance structure for styles and functionality is not seamless because the original WP template were not built with these principles in mind, but it should give you a sense of how it could work once we start building things from scratch.

I went with Handlebars templates because the Twig PHP integration is only half-baked - a bunch of build steps were broken. We are a long way from having true template-level integration anyway, so I don't think this matters.

There's no Sass support built in, though you'll see the remnants of the out-of-the-box suggested setup for scss files. I couldn't figure out how to get Sass compilation built into the watch and build pipelines in a way that made sense. And since we don't currently use CSS compilation anyway, it seemed like one too many mountains to climb at a time. So all styles are currently put into the megalithic source/css/style.css, where I've left some of the table of contents and other structure suggested by the PL default configuration. We can explore CSS precompiling as a next step.

Right now, the styleguide is totally separate from the WordPress application. The style.css stylesheet might be a good place to consider the first point of true integration. I've intentionally tried to build the demo stylesheet in such a way that it was all copy-paste from our production styles. This means that we could, theoretically, yank the copied styles out of bp-nelo/_inc/css/custom.css, and load this style.css separately as the "shared stylesheet" or something like that. In this way, we could progressively move more shared styles into the styleguide and migrate them out of the theme. Then we'd need a build step (manual or automated) that moved the styleguide CSS file into the production build at the time of deployment. Let's talk more about that as we begin to tackle implementation of the homepage redesign.

Ray and Jeremy especially, I'd be glad to hear any thoughts you have about this workflow (meager as it is).

#38 - 2021-03-10 08:51 PM - Sonja Leix

Thanks Boone, this is looking fantastic! So excited this is starting to take shape

#39 - 2021-08-16 01:15 PM - Sonja Leix

- Assignee changed from Sonja Leix to Sara Cannon

#40 - 2021-11-19 12:14 PM - Boone Gorges

- Target version changed from 1.19.0 to 1.20.0

I'm going to remove this from the milestone. I think we want to continue pursuing a style guide, but it's OK if it starts as a static document based on the Style Guide page in the Figma design documents that Sara created. A modular styleguide will become more important as we redesign more interior pages, so let's circle back around to the question of static vs dynamic styleguides as we begin that work.

Files

Commons-styleguide_121218.pdf	2.19 MB	2018-12-14	Sonja Leix
-------------------------------	---------	------------	------------