# AHG Presentation Outline

**Note: presentation slides will be available before shortly before the start of the conference.**

## Where does this story start?

To understand how we are building an accessible theme as WCAG 2.0/2.1 compliant as we can out of the box, we have to understand where we were at.

## Pre-design system

* Static web templates for longest time
* Move to Drupal around early 2010s (some WP/static websites)
* Major Drupal sites have their own theme
	+ Upwards of 60-70+ separate Drupal child themes of base wwuzen, which is a child of zen, etc.

**This needed to change**

## Other redesign goals

* Use a design system/living style guide methodology to guide web development
* Find a design system that will complement Drupal 8, as the homepage was moving to Drupal 8
* Develop a theme that will provide a solid base for specialized themes, as well as be compatible on both Drupal and non-Drupal sites

## Enter: Lando, Composer, Pattern Lab

[**Lando**](https://docs.devwithlando.io/)**:** local development environment preconfigured with recipes for Drupal, WordPress, Pantheon, and other platforms to quickly spin up locally.

**Composer**: we include a custom Composer file so users can start a Drupal 8 site that works with our patterns, as well as download recommended modules and dependencies.

[**Pattern Lab**](https://patternlab.io/)**:** both a design system and static site generator used to test components and the ways they work together to create a site. Our instance relies on Twig templating and YAML to extend components.

## Accessibility and Atomic Design

### Atoms

Breaking up our components into its smallest patterns allows us to manage accessibility down to the

By default:

* Text is significantly larger than it used to be, means more readability overall
* Color contrast is factored in, for hover/focus styles as well
* Background color differences accounted for
	+ Illustrate sm link various hover/focus styles differences with differing background colors
	+ gradschool, huxley – failure when background hover/focus was still lime green
* SVG logos have accessible markup
* Encourages proper semantics—ex. aside component

### Molecules

Where most content-based components live

* Ex. Search block—shows how atoms work together to create larger component
* Also place for components that present information
	+ Flowcharts
	+ Org charts
	+ Accordions
	+ Content switcher (tablist)

### Organisms

Even bigger components, distinguishable regions of a page, more complicated widgets

* Footers (Site footer vs. Western Footer)
* Quick Search
* Comments

### Templates

Examples of common page templates

* Majors
* Article page
* Faculty/Staff Profiles

## Yay patterns, but how do I use them?

<https://designsystem.wwu.edu/> -- hub for everything Western digital branding and guidelines

* Including accessibility
	+ Color Contrast chart: <https://designsystem.wwu.edu/color-cheat-sheets>
	+ Web Accessibility Overview: <https://designsystem.wwu.edu/web-accessibility-overview>
	+ Email Signature Generator: includes Western branded image with proper alt text embedded

Some components are built into Drupal as block types with a friendlier UI—whitespace, image links, quicksearch forms block, etc. This means complex widgets can be injected while maintaining their accessibility.

## Other D8 Accessibility

Layout Builder—formatter for creating different page layouts—2col, 3col, different widths—allows us to present content meaningfully without worrying so much about reading order or losing semantic accessibility

Example: Diversity Page (<https://www.wwu.edu/diversity/>)

## So this all went smoothly without a hitch?

Some things we learned along the way:

* If you have accessibility devs, help get the dev environment as solid as you can, as soon as you can
	+ Allows them to also work on components, without having to redo patterns once entire library is pretty much developed
* Include accessibility devs and testers in PRs for new components and templates
* Accessibility testing at the design stage—consider tab focus, heading structure, naming, etc.
* Develop for backwards capability if needed
	+ Gradschool/Huxley build script

## Thanks

* To my team for embracing and fighting for accessible web content
* To our WebTech director that supports accessibility efforts
* To you for attending this presentation!

## Any Questions?