DRUPAL Helps

LEARNING MANUAL · DRUPALCAMP CHATTANOOGA 2026

AI + Drupal Canvas for everyday site builders

What these tools actually do, where they fit in your workflow, and how to start building with them, using the free Jarvis theme.

Rod Martin

DrupalHelps.com · NavigateTomorrow, Inc.

Saturday, November 14, 2026

START HERE

Welcome

This manual goes with the 45-minute session. The slides give you the big picture. This manual gives you the details you'll need when you sit down to build: the vocabulary, the setup steps, the exact prompts from the live demo, and fixes for the errors people hit most often.

You don't need any AI or Canvas experience. If you've built a page in Drupal before, you're ready.

What you'll be able to do

- → Explain what Drupal Canvas and the Drupal AI module do, in plain language.
- → Decide when Canvas is the right tool and when to wait.
- → Connect an AI provider to Drupal safely, using the Key module.
- → Build a page by hand in Canvas, then build the same page with Canvas AI.
- → Install the Jarvis theme and its recipe on a fresh local site with DDEV.

→ Explain what an accessible theme does for ADA compliance, and what it can't do.

Session agenda

- [ ] `0:00` Why this matters now (5 min)
- [ ] `0:05` Drupal Canvas: the visual page builder (9 min)
- [ ] `0:14` Drupal AI: what it does for site builders (9 min)
- [ ] `0:23` Jarvis: bringing Canvas and AI together (6 min)
- [ ] `0:29` Live demo: build a page by hand, then with AI (12 min)
- [ ] `0:41` Your Monday plan and Q&A (4 min)

> **How to use this manual**
>
> Read Parts 1 to 4 for the concepts. Work through Part 5 at your desk to build everything yourself. Every command in this manual runs **locally**, on your own computer, inside DDEV.

CONTENTS

What's inside

Part 1 · Why now

Part 2 · Drupal Canvas

Part 3 · Drupal AI

Part 4 · Jarvis and accessibility

Part 5 · Hands-on labs

Part 6 · Troubleshooting

Part 7 · Your Monday plan

Glossary

Resources

PART 1

Why now

You don't need to write code to build an ambitious Drupal site. You need good building blocks and a faster way to put them together. In 2026, both arrived in stable form.

Where things stand (November 2026)

Project	Current version	What it means for you
Drupal Canvas	1.12.0, Sept 25, 2026	Stable and covered by the Drupal Security Team. Requires Drupal core 11.3 or newer.
Drupal AI	1.5.2, Oct 28, 2026	Stable. A 1.6 alpha exists; stick with the stable release on real sites.

Project	Current version	What it means for you
Drupal CMS	2.0, Jan 28, 2026	Ships with Canvas as the default page editor. The fastest way to see Canvas without setup.
Drupal core	11.x today; 12.0 due the week of Dec 7, 2026	Build on 11.3+ now. Jarvis is already marked compatible with 12.
Jarvis theme	3.5.2	Free theme from DrupalHelps, built for Canvas, with the AI stack wired in by a recipe.

Canvas reports over 11,800 sites in use on drupal.org. Version numbers change often, so check the project pages listed in Resources before you start a real build.

The big idea

Think of it as three layers that stack:

1. **Components** are the building blocks. Your theme provides them.

2. **Canvas** is the faster way to put the blocks on a page, with a live preview.
3. **AI** is a faster way to drive Canvas: you describe the page and it places the blocks.

Each layer only works as well as the one under it. Weak components produce weak AI pages. That's why this session ends with a theme, not just a module.

PART 2

Drupal Canvas

Canvas is a visual page builder for Drupal. You assemble pages from components and see the result as you work, instead of filling in forms and checking the front end afterward.

What it is

Canvas started life under the name *Experience Builder* and was renamed before its beta. Version 1.0 shipped in December 2025. It's a contributed module you can add to any Drupal 11.3+ site, and it's the default editor in Drupal CMS 2.

- → Drag-and-drop components with a live preview
- → Edit text right on the page, with undo
- → Reusable patterns and page regions (header, footer)
- → Content templates that lay out every node of a content type

→ Built on Drupal's structured content, revisions, and permissions

The six words to know

Component

> A building block such as a hero, card, or stat. In themes these are **Single Directory Components (SDCs)**: one folder holding the template, styles, and a YAML file that describes its settings. Example: Jarvis's `hero` component.

Props

> The settings an editor fills in: heading, image, button text, alignment. Props are what appear in the right-hand panel when you select a component. Example: the hero's `heading`, `subheading`, and `cta_text`.

Slots

> Drop zones inside a component where other components go. A three-column layout has three slots. Props are form fields; slots are empty boxes.

Patterns

> A saved group of components you can drop onto any page in one move. Good for a "speaker row" or "pricing section" you reuse.

Content templates

> A layout for every node of one content type. You map fields (title, image, body) into components once, and every blog post uses it.

Code components

> Components written in JavaScript (React) right in the browser. Powerful, and completely optional for site builders.

If you know Layout Builder

Task	Layout Builder	Canvas
Add content	Add a section, then a block, then fill a modal form	Pick a component and drop it where it goes
Preview	After you save the block	Live, as you type
Design pieces	Custom block types and fields for each one	Components come from the theme

Task	Layout Builder	Canvas
Editor guardrails	Editors can rearrange sections freely	Editors only see the props you expose
Future work	Maintained, stable	Where new site-building effort is going

Layout Builder isn't going away, and plenty of production sites depend on it. You don't need to migrate working sites. Use Canvas for new builds and new landing pages.

When to use Canvas

GREAT FIT TODAY

- → Landing and campaign pages
- → Home pages and section fronts
- → New sites on Drupal 11.3+
- → Teams that want editors to be self-sufficient

TEST FIRST

- → It's a 1.x module that ships often: pin versions in Composer
- → Requires Drupal core 11.3 or newer
- → Contributed module support is still growing
- → Try your must-have modules on a staging copy

Think about it

Pick one site you maintain. Which pages are rebuilt most often by hand? Those are your best first Canvas candidates. Which pages rely on a contributed module that you haven't tested with Canvas? Those wait.

PART 3

Drupal AI

The AI module isn't a chatbot bolted onto Drupal. It's a framework that lets Drupal talk to the AI provider you choose, with you in control of what it can do.

How it's put together

1. **Key module.** Stores your API key outside of normal configuration. It's the AI module's only hard dependency.
2. **AI core.** One settings page at `/admin/config/ai/settings` where you pick a default provider and model for each kind of job: chat, chat with tools, images, embeddings.
3. **Provider modules.** OpenAI, Anthropic, Google Gemini, AWS Bedrock, Azure, Ollama (runs on your own hardware), and many more. drupal.org lists about 95.

4. **Features.** Submodules and contributed modules that use the provider: alt text, editor tools, automators, chatbots, Canvas AI.

Because features talk to AI core instead of to a vendor, you can swap providers without rebuilding anything. Your content model stays yours.

What AI does for a site builder today

Feature	What it does	Where you'll find it
Image alt text	Suggests alt text when an image is uploaded. Editors review and fix it.	AI Image Alt Text submodule; a button on the image field
Editor assist	Rewrite, summarize, change tone, or translate inside CKEditor	AI CKEditor submodule; add the AI button to your text format toolbar

Feature	What it does	Where you'll find it
Automators	Fill fields for you: tags, summaries, meta descriptions	AI Automators submodule; a setting on each field
Chatbots	A site assistant that answers from your own content	AI Assistant API and AI Chatbot submodules, plus AI Search for retrieval
Canvas AI	Describe a section or page and get it built from your components	The `canvas_ai` submodule of Canvas; the sparkle button in the editor
Logging and moderation	A record of AI requests and a check on what goes in and out	AI Logging submodule; moderation settings for your provider

Best first win

Turn on AI alt text first. It saves editors real time and improves accessibility, as long as someone reads each suggestion before saving. AI alt text describes what's in the picture; only a person knows why the picture is on the page.

Canvas AI is a team of agents

When you type into the Canvas AI panel, an **orchestrator** reads your prompt and hands it to the agent best suited to the job.

Agent	What it does	Triggered by words like
Page builder	Places your existing components and fills in their props	"Place...", "Use...", "Add..."
Template builder	Builds a whole page layout in one pass	"Create a template for..."
Component agent	Writes a brand-new code component	"Create a component..."

Agent	What it does	Triggered by words like
Title and metadata	Writes the page title and meta description	Runs when those are empty

Watch out for the word "create." It can send your request to the component agent, which writes new code instead of using your theme's components. For beginners, that's almost never what you want.

Turning it on

- [ ] Enable the `canvas_ai` submodule that ships with Canvas.
- [ ] Connect a provider whose models support **tool calling** (current OpenAI, Anthropic, and Gemini models do).
- [ ] At `/admin/config/ai/settings`, set that provider for **Chat with tools**.
- [ ] Give your editor role the Canvas AI permission.
- [ ] Open a page in Canvas and look for the sparkle button.

Write component descriptions

Canvas AI decides which component to use by reading each component's description. Short, plain descriptions make a big difference. Set them under **Configuration › AI › Canvas AI Component Description Settings**.

```
Hero: Use once, at the very top of a page.
Big heading, one sentence of
supporting text, and one button.
3 Columns: Use to show three related items
side by side. Put a Stat or a
Card in each column.
```

Prompting Canvas AI

Talk to it like a junior site builder who knows your components but not your project.

1. **Start with "Using the page builder tool,"** so it places your components instead of inventing new ones.
2. **Say where.** "At the top of the page," "below the hero," "into each column."
3. **One section per prompt.** A series of focused prompts beats one giant prompt.

4. **Give it the real words.** Headings, button text, and who the page is for.
5. **Review, then publish.** Click each component and check the props it filled in.

Weak prompt

```
Make me a nice page about our workshop.
```

Strong prompt

```
Using the page builder tool, place a hero
at the top of the page.
Heading: "Build Your First Drupal Site".
Subheading: "A free Saturday
workshop for beginners". CTA text: "Save my
seat".
```

When a prompt doesn't work

- → **It placed the wrong component:** name the component exactly as it appears in the library.
- → **It couldn't nest components:** click the target (a column, for example) on the canvas first, then say "place a stat into this column."

- → **It made something up:** undo, and add "use only existing components" to the prompt.
- → **It stalls:** wait 60 seconds, then refresh. Long prompts send a lot of context.

Four honest realities

Same prompt, different result

> AI output isn't repeatable. Two people typing the same prompt can get different pages. Review every page before you publish.

It costs money per request

> Canvas AI can send 30,000 or more tokens of component information with each request. Disable components you don't use so there's less to send, and set a spending limit in your provider account.

Your content leaves the building

> Prompts, and anything pasted into them, go to your provider. Read its data policy before editors paste anything sensitive. If that's a problem, look at a self-hosted provider such as Ollama.

A human stays in charge

> AI drafts. People approve. Content moderation, revisions, and your editorial workflow still apply.

A simple AI policy for your team

Before you turn AI on for editors, agree on these five things. One page is enough.

- [ ] Which provider we use, and who holds the API key and billing
- [ ] What must never be pasted into a prompt (personal data, unreleased information)
- [ ] Who reviews AI-drafted pages and alt text before publishing
- [ ] Our monthly spending limit, and who gets the alert
- [ ] How we tell visitors when content was drafted with AI, if we do

PART 4

Jarvis and accessibility

Jarvis is a free Drupal theme from DrupalHelps, built for Canvas, accessible by default, with the AI stack connected by a recipe. It's what you take home from this session.

At a glance

Components	24 Single Directory Components, all editable in Canvas
Framework	Bootstrap 5.3.8, bundled
Requirements	Drupal 11.3 or newer (ready for 12), PHP 8.3 or newer

Install	One recipe builds the whole site: theme, Canvas setup, AI stack, and essentials
Live example	DrupalHelps.com runs on Jarvis
Source	github.com/imrodmartin/jarvis

What the recipe sets up

Canvas, wired

> Component configuration, five reusable patterns, content templates for Blog and Basic pages, and a component showcase page.

AI stack, configured

> AI core, the Canvas AI agents, CKEditor AI, and AI alt text, with OpenAI as the default provider. No API keys are included; you add your own.

Site essentials

> Webform, Metatag, an editorial workflow, Editoria11y, Linkit, and admin improvements.

Component reference

Components with **slots** hold other components. Everything else is filled in with props.

Component	Use it for	Slots
Section	A wrapper that holds other components	content
1 Column	A full-width row	first
2 Columns	Two side by side; stacks on mobile	first, second
3 Columns	Three across; two on tablet; stacks on mobile	first, second, third
4 Columns	Four across; two on tablet; stacks on mobile	four slots
3 Columns (custom widths)	Editor-set percentage widths; a 0% column collapses	first, second, third
Hero	Full-width banner with heading, subtext, button, and background image	

Component	Use it for	Slots
Text	Rich text with width and alignment controls	
WYSIWYG	Freeform CKEditor text, with an optional readable width	
Image	Responsive image with optional caption	
Image Overlay	Image with a colored text card over a lower corner	
Card	Card with the image on top, left, right, or behind	
Card Full Image	Half-width image edge to edge beside rich text	media, content
Person	Portrait, name, position, phone, email, link	
Stat card	Icon, big number, short description	
Timeline	Steps shown as tabs, an accordion, or a vertical rail	

Component	Use it for	Slots
Button	A single call-to-action button	
Call to Action	Centered heading, text, and one button	
Large Call to Action	Big heading, three icon rows, a button and a link	
Share	Share buttons for the current page	
Video	YouTube, Vimeo, or a video file	
Video Background	A band with video playing behind a text card	
Video with Sidebar	Video beside a title, description, and buttons	
Map	A Google map from a one-line address	

Accessible by default

Jarvis is built to WCAG 2.2 AA. That's a newer version of the standard that includes everything in WCAG 2.1 AA, the

level the ADA rule requires.

- → **Automatic overlay contrast.** When text sits on an image (heroes, cards), Jarvis adds an overlay dark enough to keep the text readable.
- → **Automatic text color.** On the two background colors, Jarvis picks black or white text for you, whichever passes.
- → **Live contrast badges.** The theme settings page checks each color pair against WCAG AA (4.5:1) as you pick, offers a one-click fix, and warns you again when you save.
- → **Editoria11y for editors.** Flags missing alt text, skipped heading levels, and vague link text while editors work.

ADA Title II: the dates that matter

The U.S. Department of Justice rule requires state and local government websites to meet WCAG 2.1 AA. An interim final rule on April 20, 2026 pushed each deadline back one year:

Who	Deadline
State and local governments serving 50,000 or more people	April 26, 2027
Smaller governments and special districts	April 26, 2028

This is general information, not legal advice. Confirm your organization's obligations with your own counsel.

> **An accessible theme is the starting line**
>
> No theme makes a site compliant by itself. Your editors keep the site accessible with every page they publish.

Editor checklist for every page

- [ ] Every meaningful image has alt text that says why it's there. Decorative images are marked decorative.
- [ ] Headings go in order (H2, then H3) and aren't used just to make text big.
- [ ] Link and button text makes sense on its own ("Register for the workshop," not "Click here").

- [ ] Videos have captions. Audio has a transcript.
- [ ] PDFs are tagged and accessible, or the content is also on the page as HTML.
- [ ] Editoria11y shows no errors before you publish.

PART 5

Hands-on labs

These labs rebuild the live demo on your own computer. Plan on about 90 minutes the first time.

Before you start

- [ ] DDEV installed and working (`ddev version` runs without errors)
- [ ] Docker Desktop, OrbStack, or another Docker provider running
- [ ] An OpenAI API key with billing turned on (or another provider's key; see Lab 2)
- [ ] About 3 GB of free disk space

Lab 1 · Build the site with DDEV

All commands run **locally**, one at a time, in this order. Copy them exactly: the quotes matter.

1. Create the project folder and start DDEV

```
mkdir jarvis-demo && cd jarvis-demo
ddev config --project-type=drupal11 --
docroot=web
ddev start
```

2. Install Drupal 11, then restart so DDEV writes its settings files

```
ddev composer create-project
"drupal/recommended-project:^11"
ddev restart
```

3. Register the three Jarvis repositories

```
ddev composer config repositories.jarvis-
theme
'{"type":"vcs","url":"https://github.com/im
rodmartin/jarvis","no-api":true}'
ddev composer config repositories.jarvis-
modules
'{"type":"vcs","url":"https://github.com/im
rodmartin/jarvis-modules","no-api":true}'
ddev composer config repositories.jarvis-
recipe
'{"type":"vcs","url":"https://github.com/im
rodmartin/jarvis-recipe","no-api":true}'
```

4. Require the recipe and the packages it needs

```
ddev composer require imrodmartin/jarvis-
recipe drupal/ckeditor5_markdown
"drupal/default_content:^2.0@beta"
"drupal/ai_media_image:^1.0@alpha"
drush/drush
```

5. Install Drupal

```
ddev drush site:install standard --account-
name=admin -y
```

6. Apply the Jarvis recipe

```
ddev drush recipe
/var/www/html/recipes/jarvis-recipe
```

7. Rebuild the cache and log in

```
ddev drush cache:rebuild
ddev launch $(ddev drush uli)
```

Three rules that prevent most errors

- → **Don't enable Canvas or the Jarvis theme before step 6.** The recipe installs both. If they're already there, it stops with a "configuration already exists" error.
- → **Use the full path in step 6.** Drush runs from `web/`, but `recipes/` is at the project root.
- → **Keep the quotes.** The `^` and `@` characters confuse the Mac's zsh shell without them.

Check your work

- [ ] The home page shows the Jarvis component showcase.
- [ ] Appearance shows Jarvis as the default theme.
- [ ] Opening the home page in Canvas shows the component library on the left.

Lab 2 · Connect an AI provider

The Jarvis recipe already points the AI module at OpenAI and expects a key named `openai`. You only need to create that key.

1. Go to **Configuration › System › Keys** (`/admin/config/system/keys`) and click **Add key**.
2. Key name: `openai`. Check that the machine name is exactly `openai`.
3. Key type: **Authentication**. Key provider: **Configuration**.
4. Paste your API key in **Key value** and save.
5. Go to `/admin/config/ai/settings` and confirm OpenAI is the default for **Chat** and **Chat with tools**.

Using Anthropic instead? Create a key with the machine name `claude`, then switch the defaults on the AI settings page to Anthropic. The "Configuration" key provider is fine on a local site. On a live site, use an environment variable or file-based key provider so the key isn't stored in your database or exported config.

Lab 3 · Build a page by hand

1. Go to **Content › Add content › Canvas page** (or open Canvas from the admin toolbar) and create a page titled "Build Your First Drupal Site."

2. From the component library, drag a **Hero** to the top of the page. Fill in the heading, subheading, and button text in the right-hand panel.
3. Drag a **3 Columns** component below the hero.
4. Drag a **Stat card** into each of the three columns. Fill in a number and a short description for each.
5. Click **Publish**, then view the page on the front end and resize your browser to see it stack on mobile.

Lab 4 · Build the same page with Canvas AI

Create a second Canvas page, open the AI panel with the sparkle button, and send these prompts one at a time. Wait for each to finish before sending the next.

Prompt 1

```
Using the page builder tool, place a hero
at the top of the page. Heading: "Build
Your First Drupal Site". Subheading: "A
free Saturday workshop for beginners". CTA
text: "Save my seat". Align it center.
```

Prompt 2

```
Using the page builder tool, place a three column component below the hero.
```

Prompt 3

```
Using the page builder tool, place a stat into each column of the three column component: "3 hours" for workshop length, "0 lines of code" required, and "24 components" to build with.
```

Prompt 4

```
Using the page builder tool, place a CTA at the bottom of the page. Heading: "Ready to start?". Button text: "Register now".
```

Compare the two pages

- [ ] Which was faster? Which was more accurate?
- [ ] Click each AI-placed component. Did it fill in every prop you asked for?
- [ ] Did it add anything you didn't ask for?

- [ ] What would you change in your component descriptions to get a better result?

Lab 5 · Watch Jarvis catch a contrast problem

1. Go to **Appearance › Settings › Jarvis** (`/admin/appearance/settings/jarvis`).
2. Under **Colors**, change the primary color to a pale yellow, such as `#FFE680`.
3. Watch the badge for "Primary vs white" turn to a fail. White button text on pale yellow is unreadable.
4. Click **Save configuration** and read the warning: Jarvis tells you the pair is below the WCAG AA minimum of 4.5:1.
5. Use the badge's one-click fix, or set the primary color back to a darker shade, and save again.
6. View your Lab 3 page as an editor and open Editoria11y. Fix anything it flags.

Reset and practice again

DDEV snapshots let you save the site and roll back after experiments. Both commands run **locally**.

```
ddev snapshot --name lab-start
ddev snapshot restore lab-start
```

PART 6

Troubleshooting

These are the problems people hit most often, with the fix for each. All commands run locally.

`ddev composer create-project` prints its help screen

> The version wasn't quoted, so the shell changed it. Run `ddev composer create-project "drupal/recommended-project:^11"` with the quotes.

site:install fails with "Call to a member function getInstallTasks() on null"

> Drush can't find database settings, so it tried to ask you for a database driver. Run `ddev restart` to make DDEV write `settings.php` and `settings.ddev.php`, then try again. If it still fails, add `--db-url=mysql://db:db@db/db` to the command.

The recipe stops with "configuration already exists"

> Canvas or the Jarvis theme was installed before the recipe. Start over from a fresh site, or restore a snapshot

> taken before you enabled them.

The recipe says the path doesn't exist

> Use the full path inside the container: `/var/www/html/recipes/jarvis-recipe`.

There's no sparkle button in Canvas

> Check that `canvas_ai` is enabled, a provider is set for **Chat with tools**, and your role has the Canvas AI permission. Then run `ddev drush cache:rebuild`.

Canvas AI spins and returns nothing

> Usually the key or billing. Check the key's machine name matches the provider setting, and that the account has credit. Look for the error at **Reports › Recent log messages**.

Canvas AI created a new component instead of using yours

> Start the prompt with "Using the page builder tool," and avoid the word "create."

PART 7

Your Monday plan

You don't need permission or a budget to try any of this. You need a local site and an afternoon.

- [ ] Spin up a local site with **DDEV**, or try Drupal CMS 2 to see Canvas out of the box.
- [ ] Apply the **Jarvis recipe** and click through the component showcase page.
- [ ] Build one real page **by hand** in Canvas before you touch AI.
- [ ] Add **one API key** and turn on AI alt text and Canvas AI.
- [ ] Write **component descriptions** so the AI knows when to use each component.

Three things to remember

1. Canvas is ready

> For landing pages and new sites on Drupal 11.3+, it's the fastest way to build in Drupal today.

2. AI drafts, you decide

> It's only as good as your components and your review.

3. Start from Jarvis

> Free, accessible by default, and wired for both. One recipe away.

REFERENCE

Glossary

Agent

> An AI helper that can take actions with tools, such as placing a component, instead of only writing text.

API key

> A secret password that lets Drupal use your account with an AI provider. Treat it like a credit card number.

Canvas

> Drupal's visual page builder, formerly called Experience Builder.

DDEV

> A free tool that runs Drupal on your own computer inside Docker containers.

Drush

> Drupal's command-line tool. With DDEV, you run it as `ddev drush`.

Editoria11y

> A Drupal module that checks pages for common accessibility problems while editors work.

Model

> The specific AI a provider offers, such as GPT-5.2 or Claude. Different models cost different amounts.

Prompt

> The instruction you type to the AI.

Provider

> The company or service running the AI model: OpenAI, Anthropic, Google, or your own server.

Recipe

> A package of modules, configuration, and content that Drupal applies in one step to add a feature or build a site.

SDC

> Single Directory Component: a theme component kept in one folder with its template, styles, and settings.

Token

> A small piece of text, roughly three quarters of a word. AI providers charge by the token.

Tool calling

> A model's ability to take actions through defined tools. Canvas AI needs it.

WCAG

> Web Content Accessibility Guidelines. The ADA Title II rule requires version 2.1, level AA.

KEEP LEARNING

Resources

Jarvis theme	github.com/imrodmartin/jarvis
Jarvis recipe	github.com/imrodmartin/jarvis-recipe
Drupal Canvas	drupal.org/project/canvas
Canvas AI docs	project.pages.drupalcode.org/canvas/ai-assistant
Drupal AI	drupal.org/project/ai
Drupal CMS	drupal.org/drupal-cms
DDEV	docs.ddev.com
ADA Title II rule	ada.gov web rule fact sheet
Community	#drupal-canvas and #ai channels on Drupal Slack
Training and help	DrupalHelps.com

Thank you, Chattanooga.

Grab Jarvis, build one page this week, and tell me what broke. That's how it gets better.

Rod Martin · DrupalHelps.com