RustNL GOSIM 2024 EUROPE

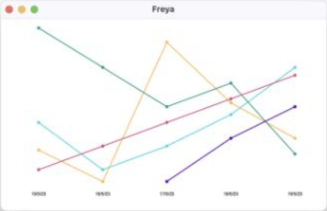
DIOXUS AI

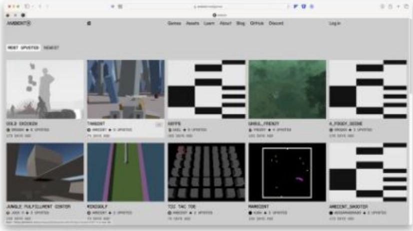
THE FUTURE OF APP DEVELOPMENT

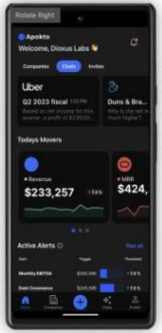
EVAN ALMLOFF, JONATHAN KELLEY

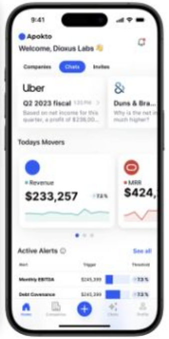
Dioxus

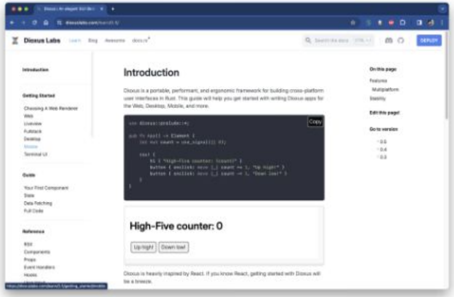
Fullstack, crossplatform, typesafe, and blazingly fast.









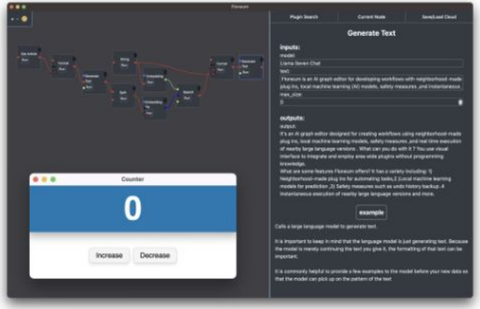


Introduction

Dioxus is a portable, performant, and ergonomic framework for building cross-platform user interfaces in Rust. This guide will help you get started with writing Dioxus apps for the Web, Desktop, Mobile, and more.

High-Five counter: 0

Dioxus is heavily inspired by React. If you know React, getting started with Dioxus will be a breeze.



Generate Text

inputs:

model:

Llama Seven Chat

text:

Floneum is an AI graph editor for developing workflows with neighborhood-made plug ins, local machine learning (AI) models, safety measures ,and instantaneous

max_size:

0

outputs:

output:

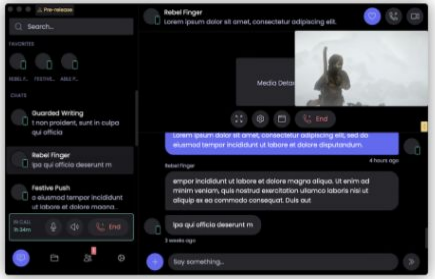
It's an AI graph editor designed for creating workflows using neighborhood-made plug ins, local machine learning models, safety measures ,and real time execution of nearby large language versions . What can you do with it ? You use visual interface to integrate and employ area wide plugins without programming knowledge.

What are some features Floneum offers? It has a variety including: 1) Neighborhood-made plug ins for automating tasks,2) Local machine learning models for prediction ,3) Safety measures such as undo history backup ,4 Instantaneous execution of nearby large language versions and more.

Calls a large language model to generate text.

It is important to keep in mind that the language model is just generating text. Because the model is merely continuing the text you give it, the formatting of that text can be important.

It is commonly helpful to provide a few examples to the model before your new data so that the model can pick up on the pattern of the text.



Rebel Finger

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Guarded Writing

t non proident, sunt in culpa qui officia

Rebel Finger

ipsa qui officia deserunt m

Festive Push

o eiusmod tempor incididunt ut labore et dolore magna

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore disputandum.

4 hours ago

empor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aut

ipsa qui officia deserunt m

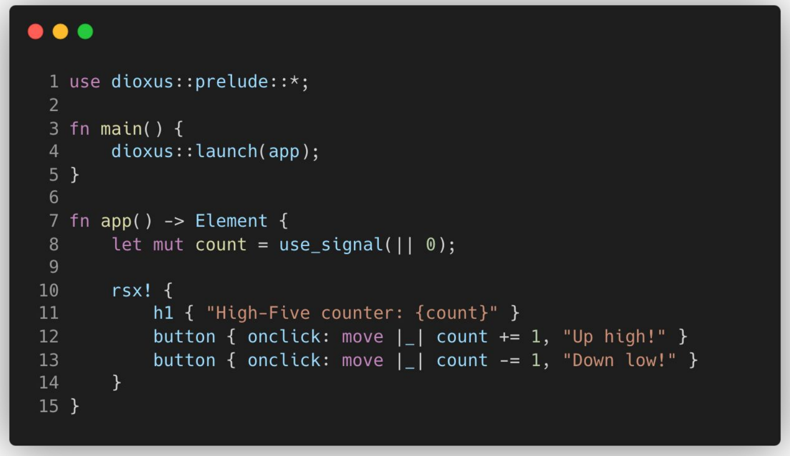
3 weeks ago

Dioxus: web, desktop, and mobile in one codebase

- Inspired by React and SolidJS, based on signals, HTML and CSS

- Up-and-running in seconds with integrated hotreloading and bundling

- Executing natively across six platforms with a single codebase.

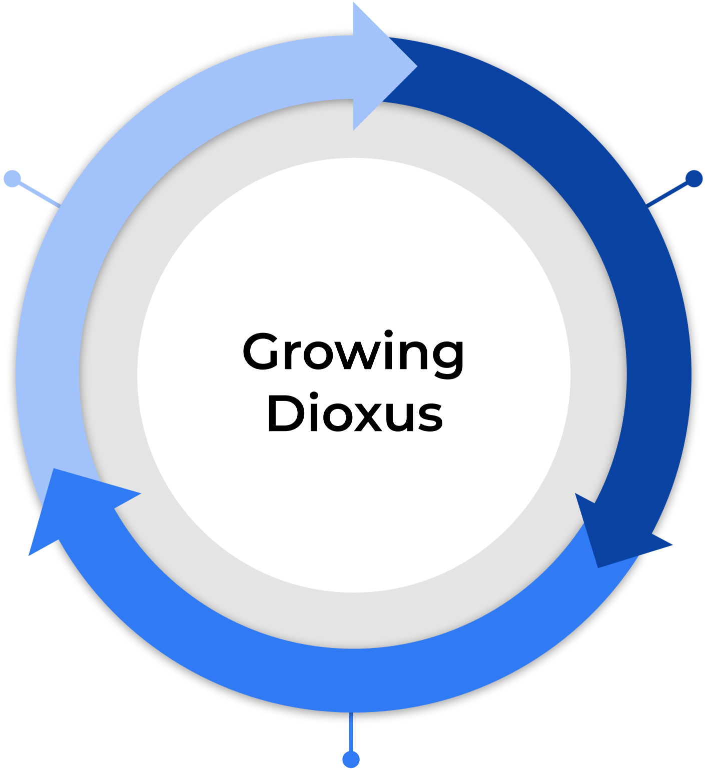


```rust
use dioxus::prelude::*;

fn main() {
    dioxus::launch(app);
}

fn app() -> Element {
    let mut count = use_signal(|| 0);

    rsx! {
        h1 { "High-Five counter: {count}" }
        button { onclick: move |_| count += 1, "Up high!" }
        button { onclick: move |_| count -= 1, "Down low!" }
    }
}
```



Larger ecosystem

More projets result in more libraries, docs, code, and examples.

Improve developer productivity

Ship things like hotreloading, autoformatting, bundling, signals

Ship projects faster

Better tooling means more projects get shipped!

Strategy for increasing developer productivity:

1. Decrease quantity of boilerplate code
2. Improve reliability
3. Shorten iteration cycles

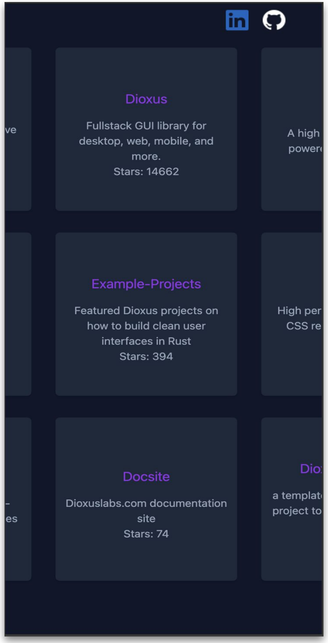
“Computer, make me an app”

A perfect Dioxus Copilot 🌈🪄✨

1. Help build and improve UIs with minimal friction
2. Automatically perform quality assurance by running tests and giving feedback
3. Improve existing code in-place, fixing bugs and iterating on things like accessibility, layout, styling, and content

Automatic refactors, accessibility, styling, new pages, forms, backend, database queries, 3rd party libraries, responsiveness, and more!

A perfect Dioxus Copilot 🌈🪄✨



ve

Dioxus

Fullstack GUI library for desktop, web, mobile, and more.
Stars: 14662

A high
powere

Example-Projects

Featured Dioxus projects on how to build clean user interfaces in Rust
Stars: 394

High per
CSS re

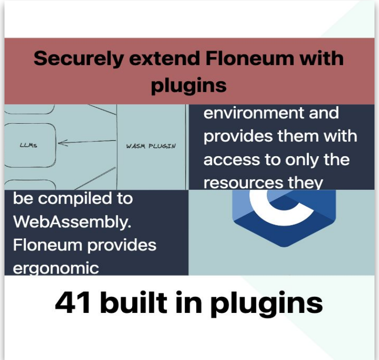
-
es

Docsite

Dioxuslabs.com documentation site
Stars: 74

Dio

a templat
project to

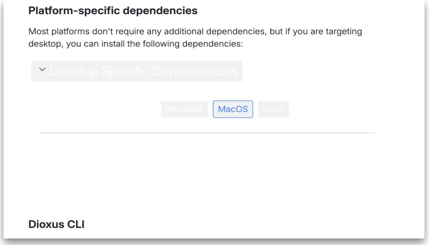


Securely extend Floneum with plugins

environment and provides them with access to only the resources they

be compiled to WebAssembly. Floneum provides ergonomic

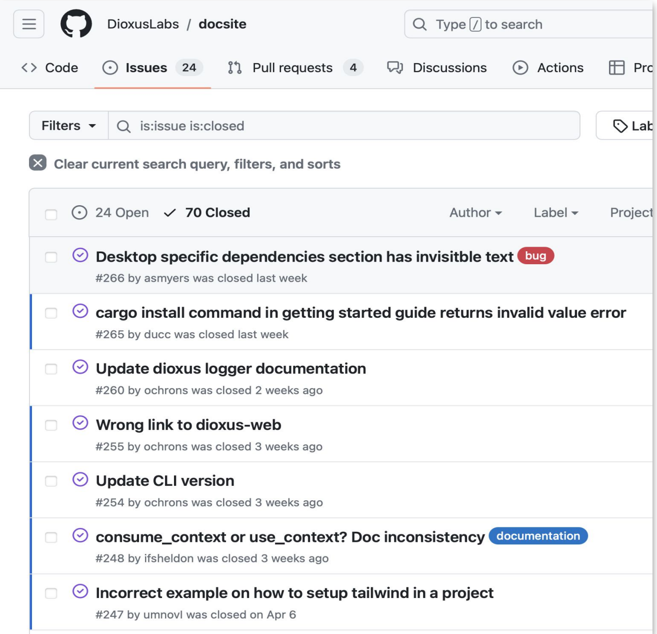
41 built in plugins



Platform-specific dependencies

Most platforms don't require any additional dependencies, but if you are targeting desktop, you can install the following dependencies:

Dioxus CLI



is:issue is:closed

Clear current search query, filters, and sorts

24 Open ✓ 70 Closed

- Desktop specific dependencies section has invisitble text `bug`
  #266 by asmyers was closed last week
- cargo install command in getting started guide returns invalid value error
  #265 by ducc was closed last week
- Update dioxus logger documentation
  #260 by ochrons was closed 2 weeks ago
- Wrong link to dioxus-web
  #255 by ochrons was closed 3 weeks ago
- Update CLI version
  #254 by ochrons was closed 3 weeks ago
- consume_context or use_context? Doc inconsistency `documentation`
  #248 by ifsheldon was closed 3 weeks ago
- Incorrect example on how to setup tailwind in a project
  #247 by umnovl was closed on Apr 6

Hard requirement: local-first

- LLMs are increasingly capable on lower-end hardware. Models like Phi-3 can run on a phone!
- Local-first LLMs save us from running any infrastructure - we want to improve Dioxus, not run servers!
- Local-first increases our “hackability” - meaning the OSS community can improve our work without much cost overhead
- We can roll out Dioxus AI via our CLI tools without requiring internet access or sending anyone’s code to our servers

Phi-3-mini-128k-instruct
Required mem: 1.8 GB @ 4bit

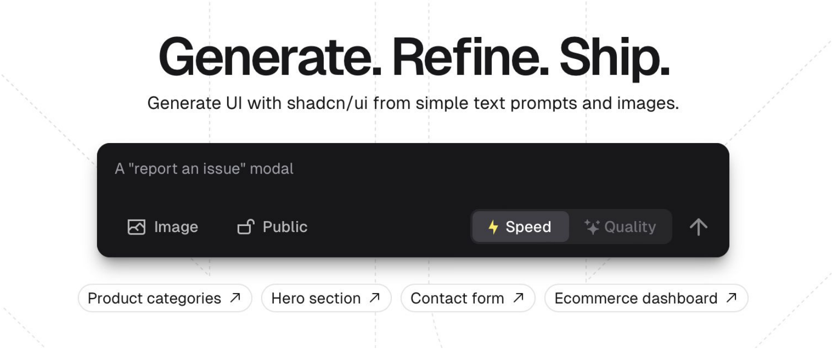
Llama 3 8b
Required mem: 5.7 GB @ 4bit

“Can a robot take a blank canvas and turn it into a masterpiece?”

“Can you?”

So much potential in automatic UI generation!

- Generate your team’s own component library
- Iterate on styles, content, and layout
- Generate mocks to present your app with fake user data
- Automating boring, tedious UI code like forms and login flows



Generate. Refine. Ship.

Generate UI with shadcn/ui from simple text prompts and images.

A "report an issue" modal

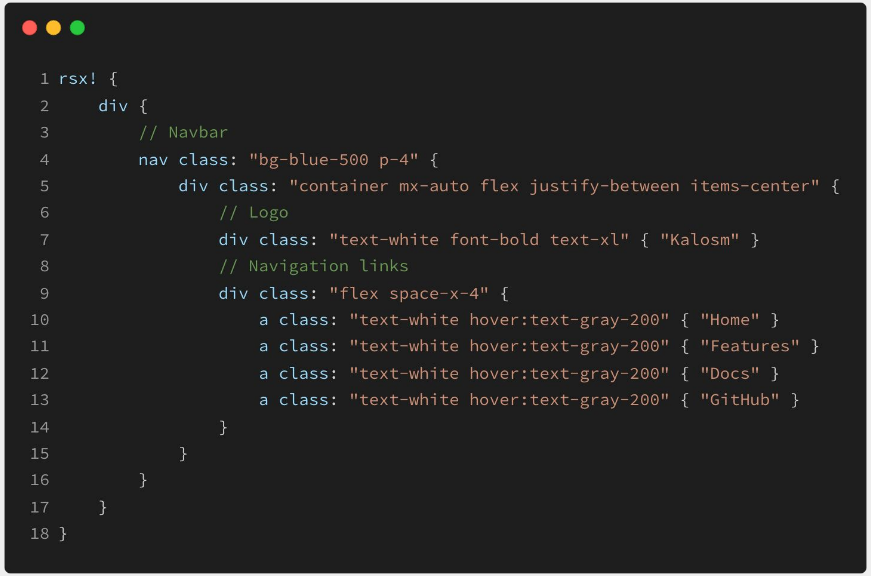
See: Vercel’s v0.dev

Can plain Llama 70B generate Dioxus?

Nope, not really.

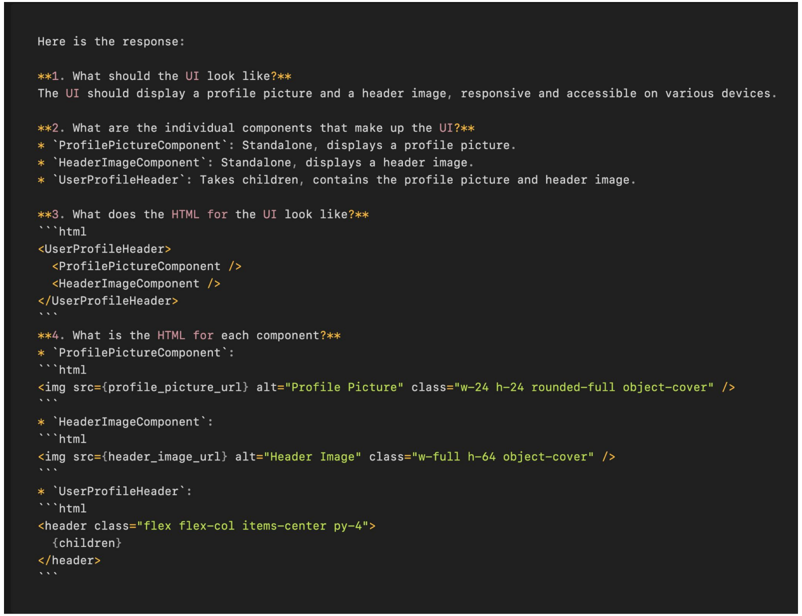
Llama has never seen Dioxus code before and gets confused when asked to generate it. Llama is still confused even when we give it our docs!

But it has seen a lot of HTML!



```
rsx! {
    div {
        // Navbar
        nav class: "bg-blue-500 p-4" {
            div class: "container mx-auto flex justify-between items-center" {
                // Logo
                div class: "text-white font-bold text-xl" { "Kalosm" }
                // Navigation links
                div class: "flex space-x-4" {
                    a class: "text-white hover:text-gray-200" { "Home" }
                    a class: "text-white hover:text-gray-200" { "Features" }
                    a class: "text-white hover:text-gray-200" { "Docs" }
                    a class: "text-white hover:text-gray-200" { "GitHub" }
                }
            }
        }
    }
}
```

Can Llama 70B generate HTML?

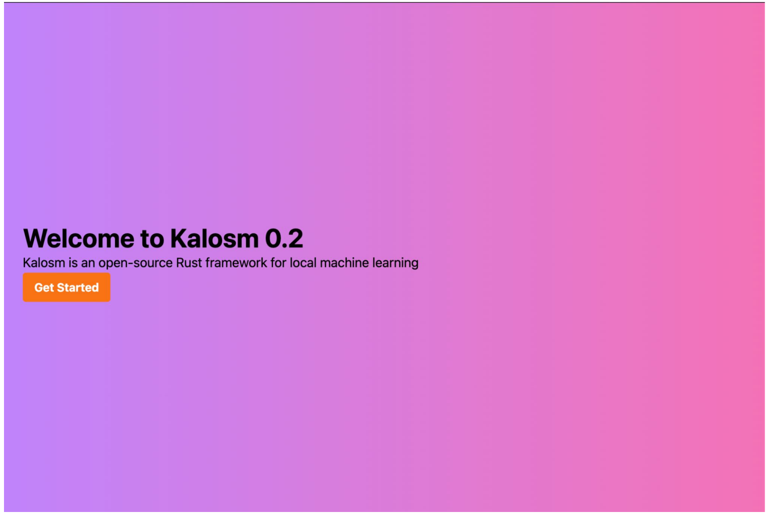


~~~
Here is the response:

**1. What should the UI look like?**
The UI should display a profile picture and a header image, responsive and accessible on various devices.

**2. What are the individual components that make up the UI?**
* `ProfilePictureComponent`: Standalone, displays a profile picture.
* `HeaderImageComponent`: Standalone, displays a header image.
* `UserProfileHeader`: Takes children, contains the profile picture and header image.

**3. What does the HTML for the UI look like?**
```html
<UserProfileHeader>
  <ProfilePictureComponent />
  <HeaderImageComponent />
</UserProfileHeader>
```
**4. What is the HTML for each component?**
* `ProfilePictureComponent`:
```html
<img src={profile_picture_url} alt="Profile Picture" class="w-24 h-24 rounded-full object-cover" />
```
* `HeaderImageComponent`:
```html
<img src={header_image_url} alt="Header Image" class="w-full h-64 object-cover" />
```
* `UserProfileHeader`:
```html
<header class="flex flex-col items-center py-4">
  {children}
</header>
```
~~~



Welcome to Kalosm 0.2

Kalosm is an open-source Rust framework for local machine learning

Can Llama 70B generate HTML?

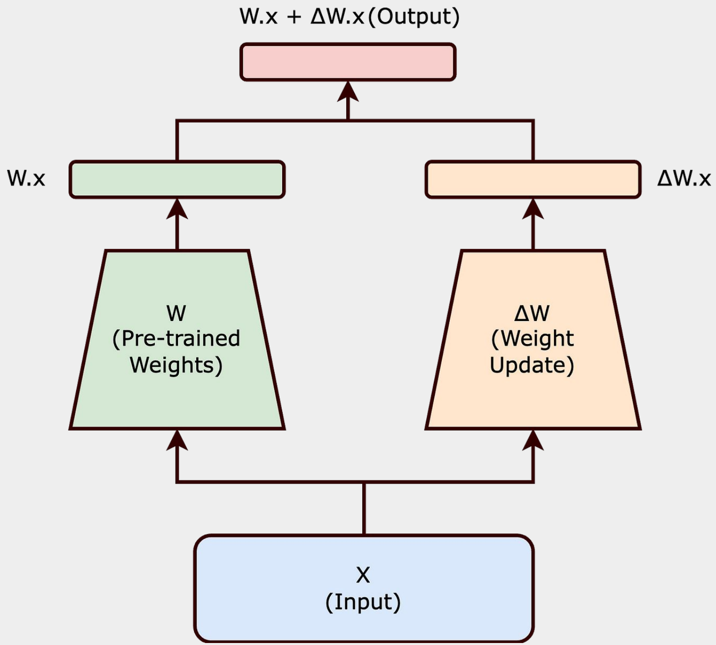
Yes!

But not very well…

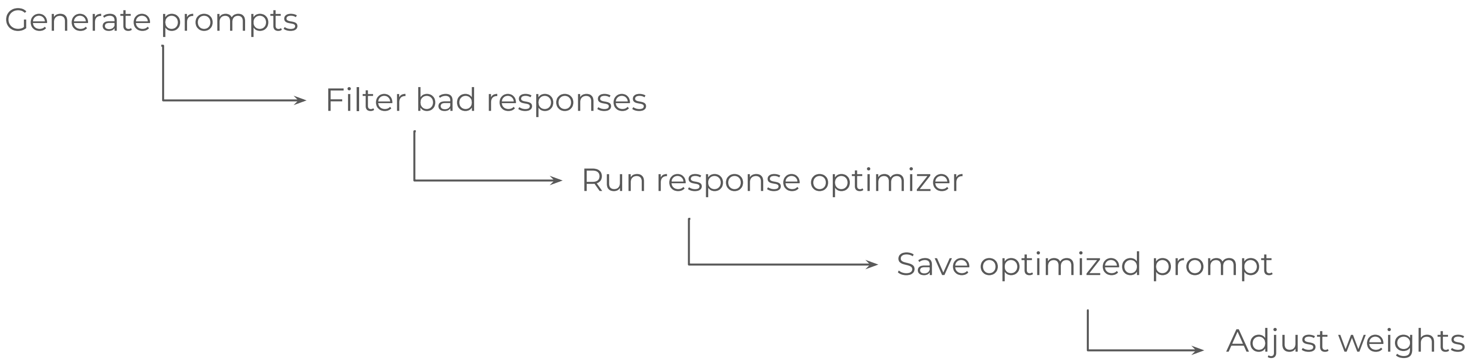
- Doesn’t always follow the output format
- Generating components that are never used, used incorrectly, or contain duplicates
- Embed javascript into JSX
- Generating noisy SVGs
- A correct output less than 20% of the time, and usable less than 10%

Fine-tuning to the rescue!

- Base Llama might not be too smart for our use case, but we can make it smarter!
- LoRA (low rank adaptation) makes it possible refine even the largest of LLMs
- We can retrain Llama using a large dataset of optimized prompts.
- We somehow need to synthesize a large, high quality dataset of prompts!



Fine Tuning using a Synthetic Dataset



Generate prompts

Filter bad responses

Run response optimizer

Save optimized prompt

Adjust weights

Technique: Filter out subpar responses

- Programmatically filter out low-quality responses from the baseline model
- Ensure the component can be parsed and that it'll generate an output
- Perform integrity and quality checks
- Lightly rewrite components for readability

Can we parse the output?

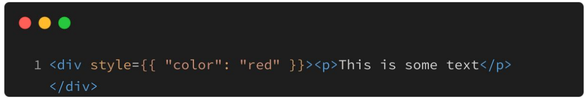
Will the component compile?

Does it answer all the questions we ask about the UI?

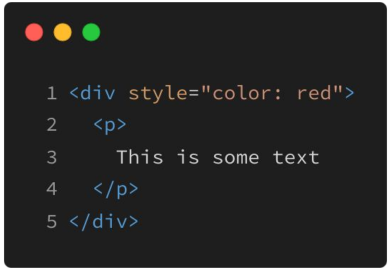
Are the components actually used?

Are there any problematic elements?

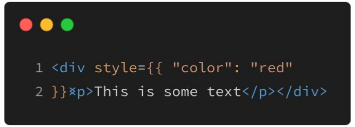
Technique: Optimize responses with normalization



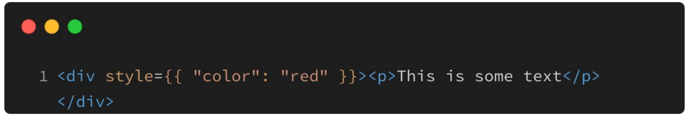
```
<div style={{ "color": "red" }}><p>This is some text</p>
</div>
```



```
<div style="color: red">
  <p>
    This is some text
  </p>
</div>
```



```
<div style={{ "color": "red"
}}><p>This is some text</p></div>
```



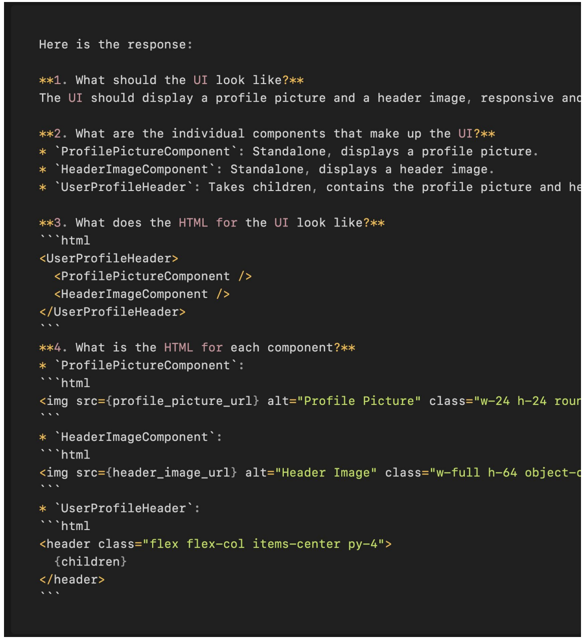
```
<div style={{ "color": "red" }}><p>This is some text</p>
</div>
```

Multiple functionally identical code snippets

A single normalized HTML output

Technique: Response compression for performance

Old: 247 tokens



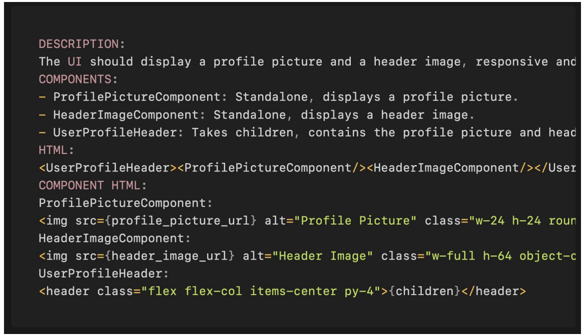
~~~
Here is the response:

**1. What should the UI look like?**
The UI should display a profile picture and a header image, responsive and

**2. What are the individual components that make up the UI?**
* `ProfilePictureComponent`: Standalone, displays a profile picture.
* `HeaderImageComponent`: Standalone, displays a header image.
* `UserProfileHeader`: Takes children, contains the profile picture and he

**3. What does the HTML for the UI look like?**
```html
<UserProfileHeader>
  <ProfilePictureComponent />
  <HeaderImageComponent />
</UserProfileHeader>
```
**4. What is the HTML for each component?**
* `ProfilePictureComponent`:
```html
<img src={profile_picture_url} alt="Profile Picture" class="w-24 h-24 rour
```
* `HeaderImageComponent`:
```html
<img src={header_image_url} alt="Header Image" class="w-full h-64 object-c
```
* `UserProfileHeader`:
```html
<header class="flex flex-col items-center py-4">
  {children}
</header>
```
~~~

New: 161 tokens



```
DESCRIPTION:
The UI should display a profile picture and a header image, responsive and
COMPONENTS:
- ProfilePictureComponent: Standalone, displays a profile picture.
- HeaderImageComponent: Standalone, displays a header image.
- UserProfileHeader: Takes children, contains the profile picture and head
HTML:
<UserProfileHeader><ProfilePictureComponent/><HeaderImageComponent/></User
COMPONENT HTML:
ProfilePictureComponent:
<img src={profile_picture_url} alt="Profile Picture" class="w-24 h-24 roun
HeaderImageComponent:
<img src={header_image_url} alt="Header Image" class="w-full h-64 object-c
UserProfileHeader:
<header class="flex flex-col items-center py-4">{children}</header>
```

35% performance boost!

~**10000** prompts generated

~**8000** followed our format

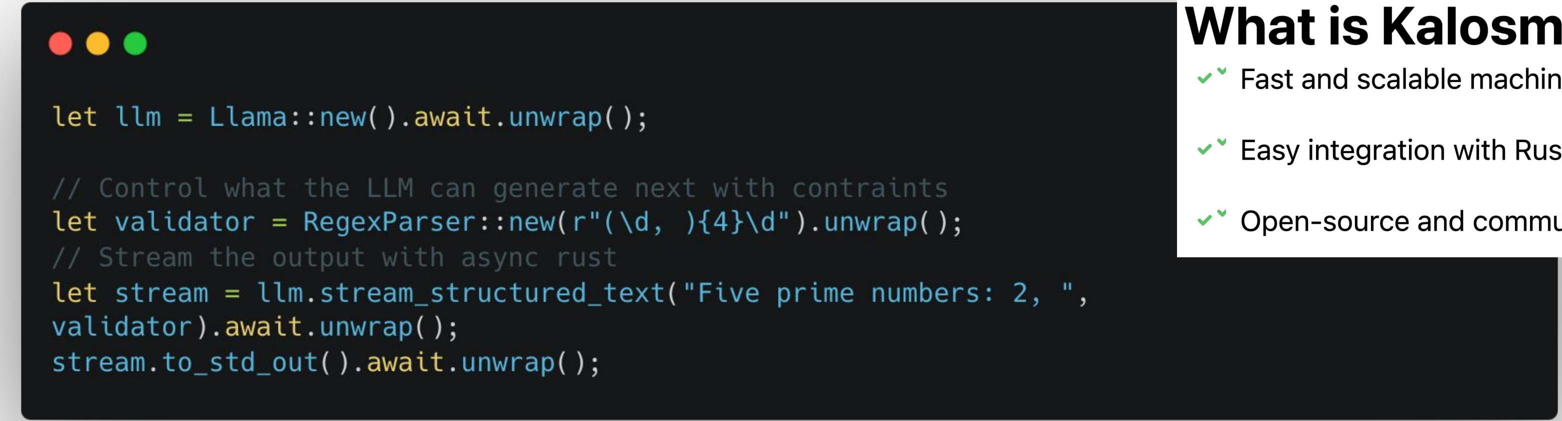
~**1500** passed quality filters

UI generation accuracy jumped from **10%** to **20%!**

Inference

Inference with Kalosm

- Quantized models that work with text, audio and image data
- Constrained generation to precisely control model outputs
- Context extraction: Webpages, PDFs, RSS, or even text from live transcription
- Integrations: Headless browser, surreal DB, arroy vector database



```rust
let llm = Llama::new().await.unwrap();

// Control what the LLM can generate next with contraints
let validator = RegexParser::new(r"(\d, ){4}\d").unwrap();
// Stream the output with async rust
let stream = llm.stream_structured_text("Five prime numbers: 2, ",
validator).await.unwrap();
stream.to_std_out().await.unwrap();
```

What is Kalosm?

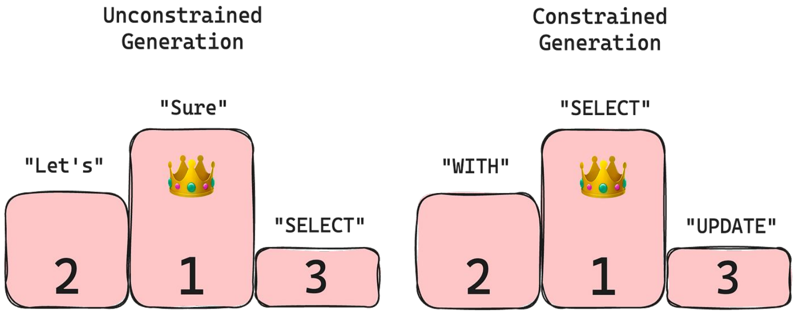
- ✓ Fast and scalable machine learning
- ✓ Easy integration with Rust ecosystem
- ✓ Open-source and community-driven

Why Kalosm: Structured Generation

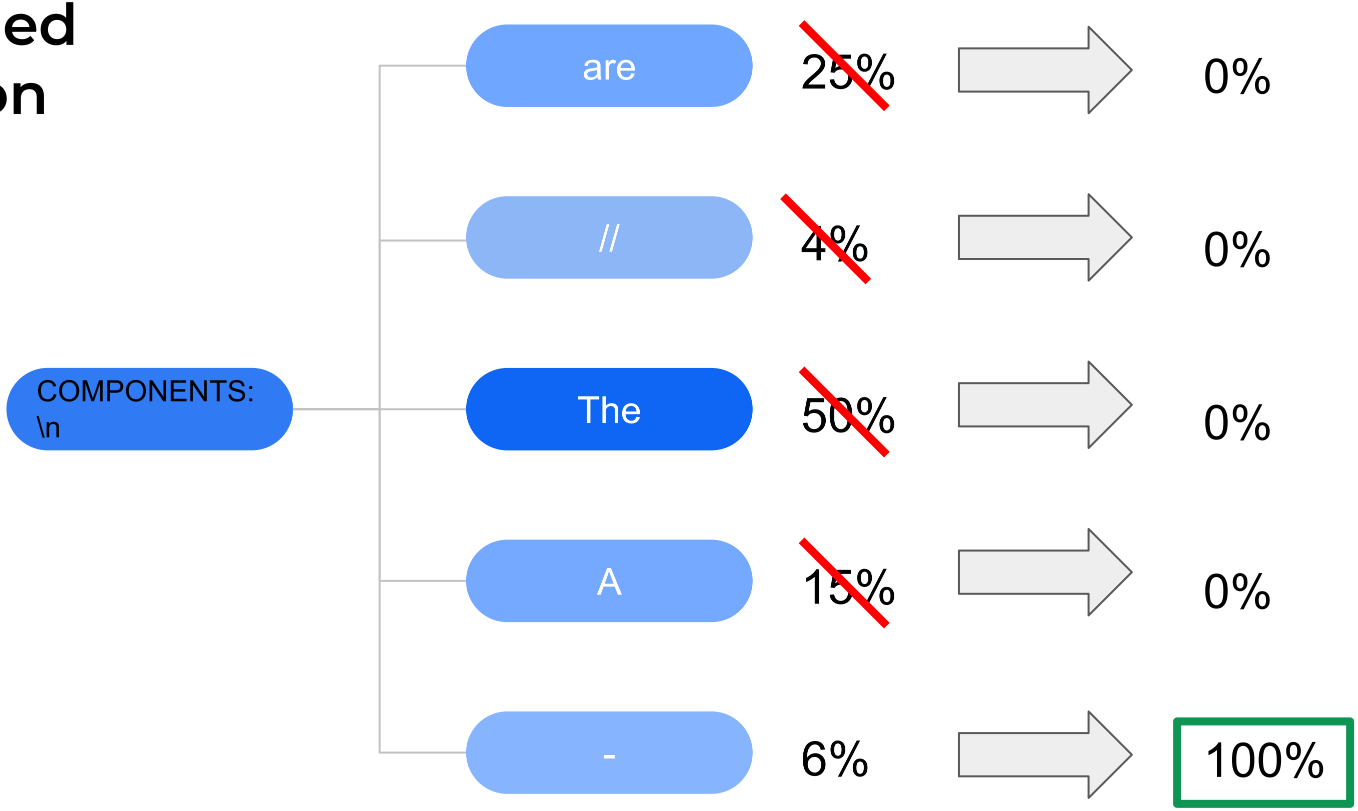
How do we *guarantee* that the model outputs code that *will* compile?

- Small models are especially prone to hallucination and “runaway” over large sequences

- Programmatically influence the output of the model depending on the state of the streaming HTML parser - Rust is really good at this! 🦀



Constrained Generation



Check Constraints

COMPONENTS:
\n

are ~~25%~~ → 0%

// ~~4%~~ → 0%

The ~~50%~~ → 0%

A ~~15%~~ → 0%

- 6% → 100%

Improves model format accuracy from **20%** to **90%!**

10 times smaller model performs 5 times better

About 2s-10s per component in the response

Constrained generation is powerful!

<|Generative UI Demo|>

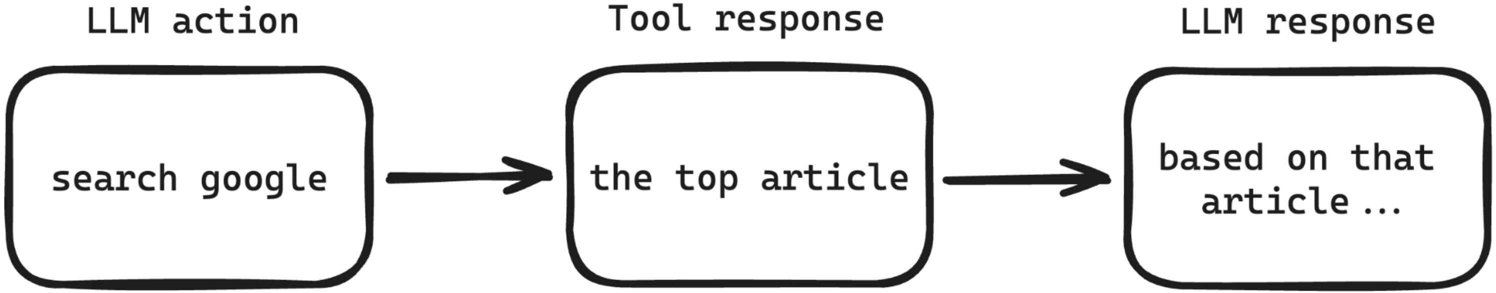
“Engineering, status report!”

“Your app could be improved, sir!”

Kalosm Agent Tools

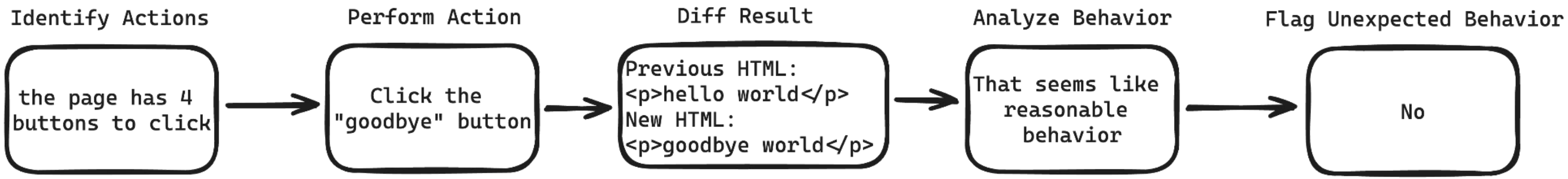
Agents combine:

- Instructions about the task
- Constraints to follow a consistent format
- Context from the environment to ground the model’s responses in reality



Automated Quality Assurance with Agents

- A very small language model (phi 3) can understand HTML
- Dioxus knows what elements are interactive
- Take actions on your site and flag unexpected behavior



Identify Actions

the page has 4 buttons to click

Perform Action

Click the "goodbye" button

Diff Result

Previous HTML:
<p>hello world</p>
New HTML:
<p>goodbye world</p>

Analyze Behavior

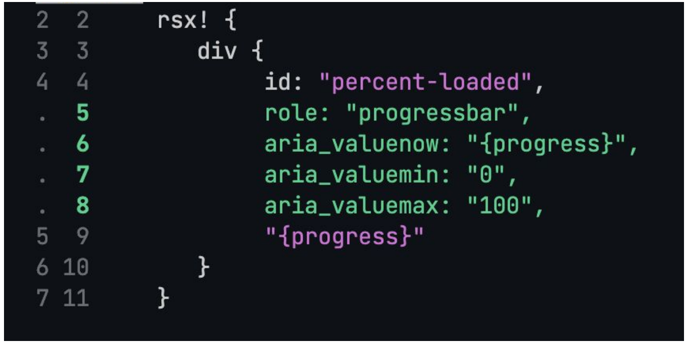
That seems like reasonable behavior

Flag Unexpected Behavior

No

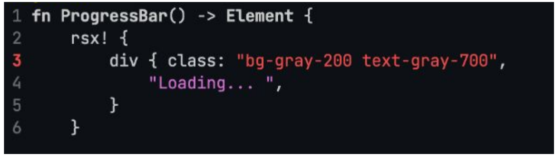
<| Kalosm Agent Demo |>

Automatically improve code in-place

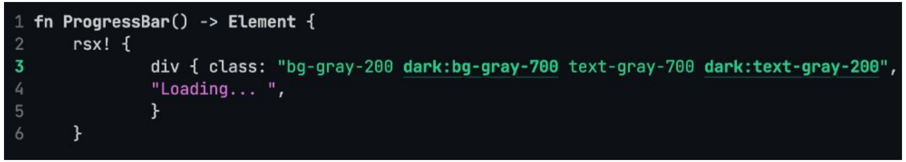


```
rsx! {
   div {
        id: "percent-loaded",
        role: "progressbar",
        aria_valuenow: "{progress}",
        aria_valuemin: "0",
        aria_valuemax: "100",
        "{progress}"
   }
}
```

Improve accessibility for screen readers



```
fn ProgressBar() -> Element {
    rsx! {
        div { class: "bg-gray-200 text-gray-700",
            "Loading... ",
        }
    }
```



```
fn ProgressBar() -> Element {
    rsx! {
            div { class: "bg-gray-200 dark:bg-gray-700 text-gray-700 dark:text-gray-200",
            "Loading... ",
            }
    }
```

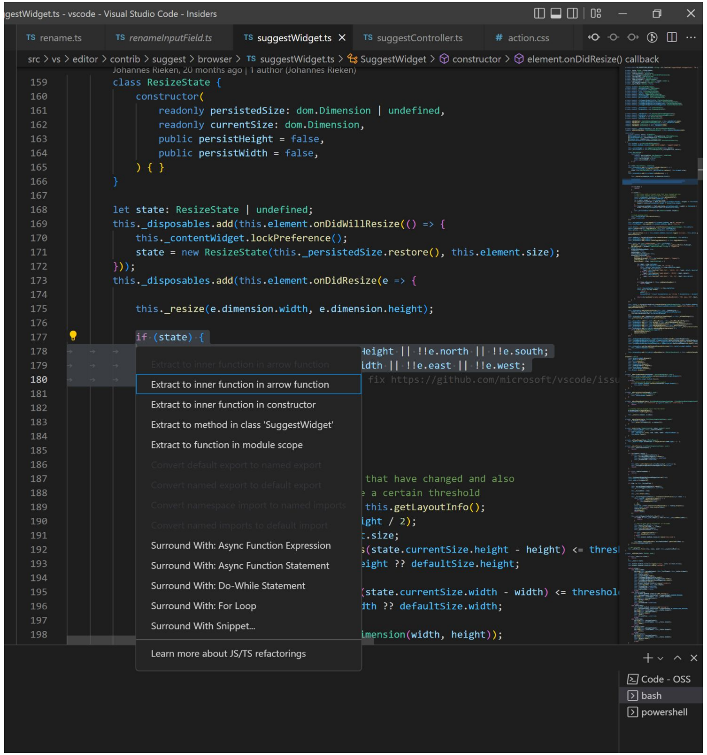
Add darkmode and media queries

Suggested refactors

Understand the codebase and dynamically suggest VSCode “magic hints”

Perform complex refactors like:

- Hoisting state into parent components
- Adding keys and other optimizations
- Moving global state to context
- Extracting elements into components
- Organizing project files


gestWidget.ts - vscode - Visual Studio Code - Insiders
TS rename.ts
TS renameInputField.ts
TS suggestWidget.ts
TS suggestController.ts
# action.css
src > vs > editor > contrib > suggest > browser > TS suggestWidget.ts > SuggestWidget > constructor > element.onDidResize() callback
Johannes Rieken, 20 months ago | 1 author (Johannes Rieken)
Extract to inner function in arrow function
Extract to inner function in arrow function
Extract to inner function in constructor
Extract to method in class 'SuggestWidget'
Extract to function in module scope
Convert default export to named export
Convert named export to default export
Convert namespace import to named imports
Convert named imports to default import
Surround With: Async Function Expression
Surround With: Async Function Statement
Surround With: Do-While Statement
Surround With: For Loop
Surround With Snippet...
Learn more about JS/TS refactorings
Code - OSS
bash
powershell

```
class ResizeState {
    constructor(
        readonly persistedSize: dom.Dimension | undefined,
        readonly currentSize: dom.Dimension,
        public persistHeight = false,
        public persistWidth = false,
    ) { }
}

let state: ResizeState | undefined;
this._disposables.add(this.element.onDidWillResize(() => {
    this._contentWidget.lockPreference();
    state = new ResizeState(this._persistedSize.restore(), this.element.size);
}));
this._disposables.add(this.element.onDidResize(e => {

    this._resize(e.dimension.width, e.dimension.height);

    if (state) {
                                    Height || !!e.north || !!e.south;
                                    idth || !!e.east || !!e.west;
                                     fix https://github.com/microsoft/vscode/issu







                                     that have changed and also
                                    e a certain threshold
                                     this.getLayoutInfo();
                                    ight / 2);
                                    t.size;
                                    s(state.currentSize.height - height) <= thres
                                    eight ?? defaultSize.height;

                                    (state.currentSize.width - width) <= threshol
                                    dth ?? defaultSize.width;

                                    imension(width, height));
```

What’s next?

- Dioxus CLI + VSCode extension integration
- Interactive UI generation
- Automated UX testing
- Fine-tuning Phi-3 and Llama 70b
- Generic implementation for other frameworks
- Shipping to production!

`https://github.com/dioxuslabs/dioxus-ai`

THANK YOU

GOSIM 2024 EUROPE