

Hello

Table of Contents

About	2
Projects	3
Devlog	4
2026-06-14 Football Field	4
2026-06-10 GHC WASM Example	4
2026-06-10 Reactive Banana Integration	5
2026-06-07 Reader Effect	6
2026-06-07 Effectful Example	7
2026-06-01 Effectful	8
2026-05-31 Weather System	8
2026-05-31 RSS	8
2026-05-28 Meow	9

About

Welcome to my homepage. Here you can find my devlog.

I think browser engines are too complex. So, this website is a PDF document.

There is an RSS feed at [./rss.xml](#).

Projects

[litmark/](#)

a portable literate-programming tangler
written in Fennel

[mill.py/](#)

a Markdown interface to LLMs written in
Python

Devlog

2026-06-14 Football Field

Hello, world cup!

62



Figure 1. Football field

2026-06-10 GHC WASM Example

<https://tildegit.org/unworriedsafari/ghc-wasm-hello>

2026-06-10 Reactive Banana Integration

I'm integrating Effectful with Reactive Banana. It doesn't appear to be super straightforward with Reactive Banana's MomentIO monad. The following code works at least.

```
withEffToIO (ConcUnlift Ephemeral
             Unlimited) $ \runInIO -> do
  network <- compile $ do
    eDraw <- fromAddHandler drawAddHandler
    eDeltaTime <- mapEventIO
      (\_ -> (*1e-3) <$>
        runInIO deltaTime)
      eDraw
    bGameState <- accumb gs $ pure stepTime
      <@> eDeltaTime
  join . liftIO . runInIO $
    Devtools.sceneMomentIO d
      fireGameState fireSceneSwitch
      bGameState
```

2026-06-07 Reader Effect

Removed the `p`-parameter by using the Reader effect. Also, eliminated the use of `Maybe` by putting the IO action itself in the state.

```
evalState (return () :: IO ()) $ do
  (drawAddHandler, fireDraw) <-
    liftIO newAddHandler

  setup $ do
    h <- windowHeight
    w <- windowWidth
    createWebGLCanvas w h

    font <- loadFont "/static/monogram.ttf"
    sheet <- loadImage "/static/Cat_Grey.png"
    put $ fireDraw (font, sheet)

  draw $ get >=> liftIO
```

2026-06-07 Effectful Example

Another example of Effectful, distributing a single state over multiple JS callbacks.

```
evalState (Nothing :: Maybe Assets) $ do
  setup p $ do
    h <- windowHeight p
    w <- windowWidth p
    createWebGLCanvas p w h
    font <- loadFont
      "/static/monogram.ttf" p
    sheet <- loadImage
      "/static/Cat_Grey.png" p
    put $ Just (font, sheet)

  (drawAddHandler, fireDraw) <-
    liftIO newAddHandler

  draw p $ do
    assets <- get @(Maybe Assets)
    mapM_ (liftIO . fireDraw) assets
```

2026-06-01 Effectful

An example that shows how to use the Effectful library: <https://github.com/beardiff/lambda-library/blob/main/app-effectful/Main.hs>

2026-05-31 Weather System

I've always felt like the Zelda world is somehow more alive than the GTA world even though there are a lot more characters in the GTA world.

Perhaps it's the weather system. In Zelda the weather itself affects Link's health. It also has lunar cycles.

2026-05-31 RSS

There is now an RSS feed at `./rss.xml!`

2026-05-28 Meow

```
Haskell kitty says meow.
```



Figure 2. Meow!

Cat asset

```
https://bowpixel.itch.io/meow-cat-85-  
animation
```

Font asset

```
https://datagoblin.itch.io/monogram
```