

What I learnt building a web app with Go and HTMX

Introduction

- Haseeb Majid
 - Backend Software Engineer at Nala
 - <https://haseebmajid.dev>
- Loves cats 
- Avid cricketer  #BazBall

Who is this for?

- Backend Developers

```
> 2 + 2
```

```
< 4
```

```
> "2" + "2"
```

```
< "22"
```

```
> 2 + 2 - 2
```

```
< 2
```

```
> "2" + "2" - "2"
```

```
< 20
```



Story Time



Banter Bus



Enter a nickname and
choose an avatar

GracefulBlushOrangutan

PLAY



English (GB)



Frontend Trace ID: Loading...

Click to copy



Banter Bus

Room Code

5GOKT



HostPlayer

NOT READY

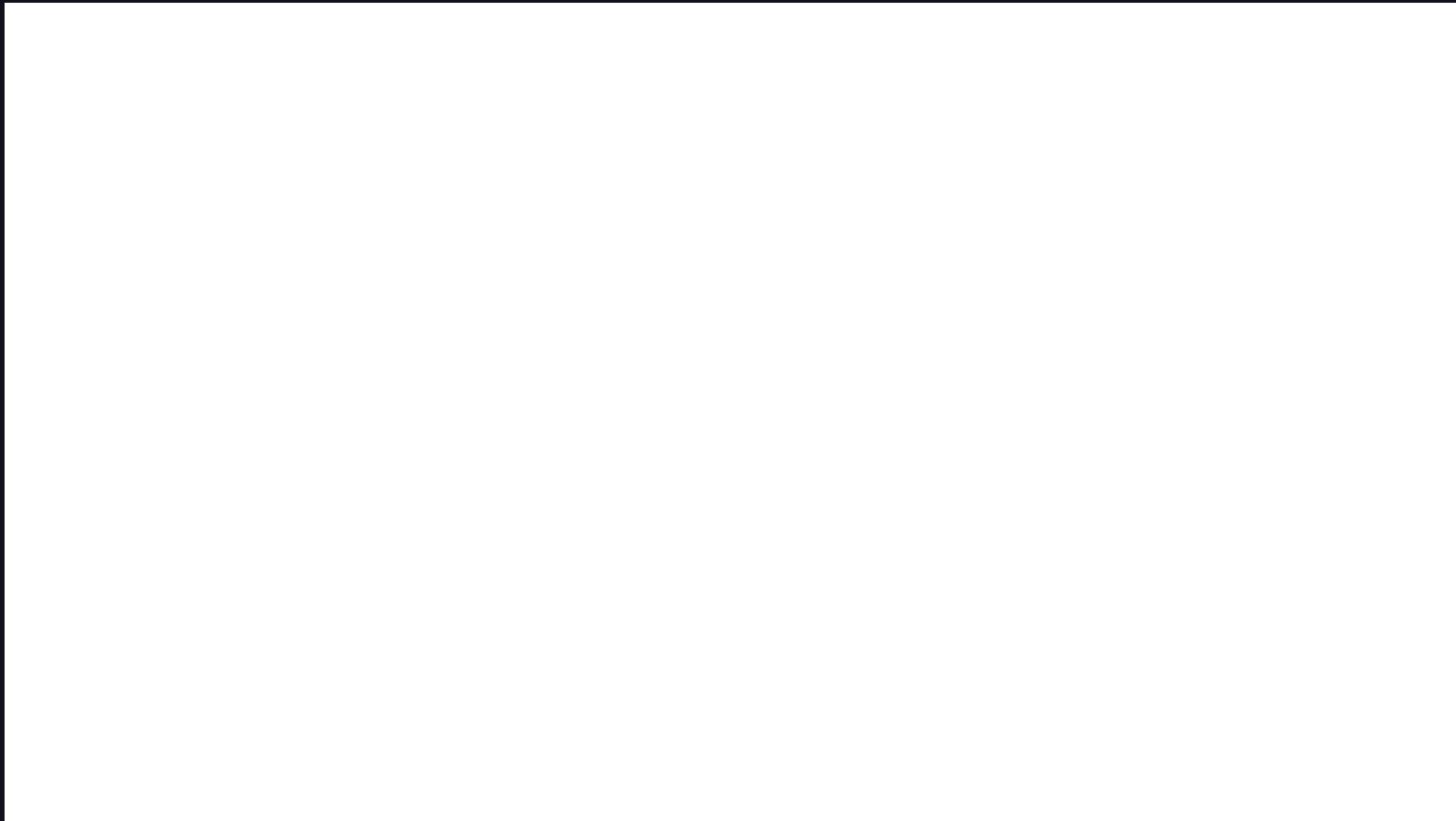


OtherPlayer0



NOT READY

Ready

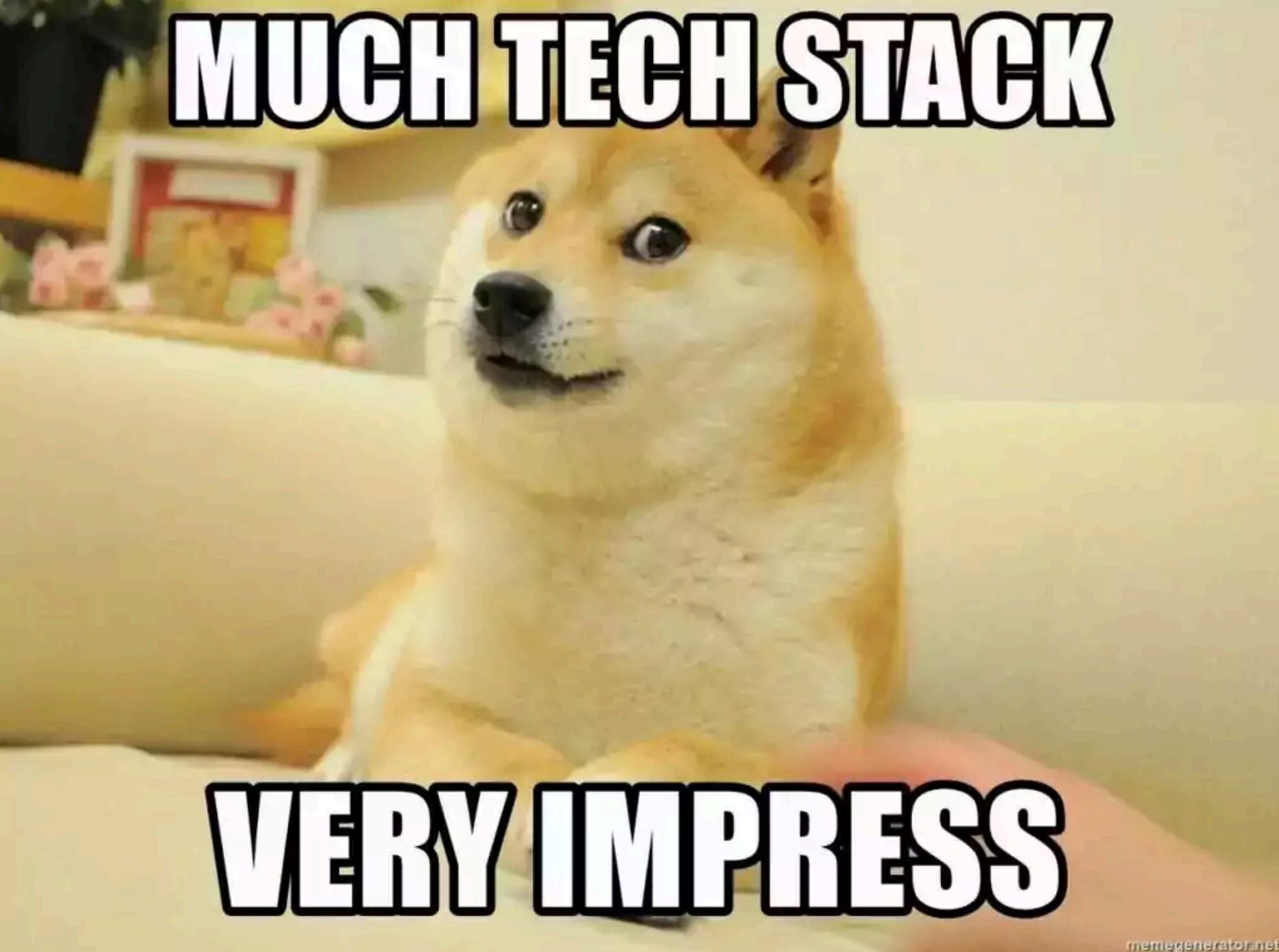


Tech Stack (Backend)

- Go
- Postgres
- Templ

Tech Stack (Frontend)

- HTMX
- TailwindCSS
- AlpineJS



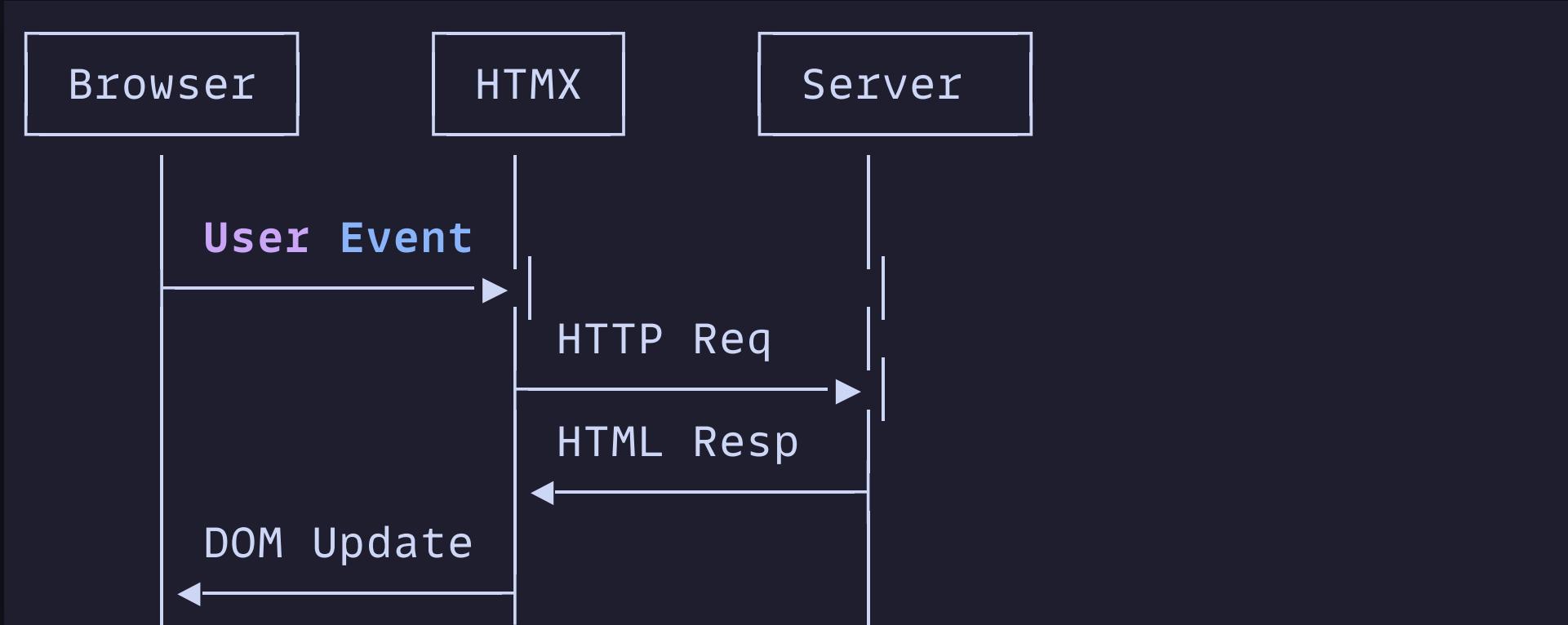
MUCH TECH STACK

VERY IMPRESS

HTMX

**javascript fatigue:
longing for a hypertext
already in hand**

– htmx.org



Banter Bus

Nickname
Haseeb

Room Code
ABC12

Start Join

Test Name:
Enter test name...

ting for plausible.haseebmajid.dev...

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Filter URLs

	Method	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings	Stack Trace
s	GET	127.0.0.1:7331	ws	htmx.js:428 (websocket)	plain	271 B	0 B				Disable Cache	No Throttling	

No messages for this request

1 request | 0 B / 271 B transferred | Finish: 2.45 s

No messages

Attribute	Purpose	Example
hx-get	GET request	hx-get="/users"
hx-post	POST request	hx-post="/users"
hx-trigger	Event trigger	hx-trigger="click"
hx-target	Target element	hx-target="#result"
hx-swap	Swap strategy	hx-swap="innerHTML"

Swap Strategies

```
←!— Replace inner content →  
<div hx-swap="innerHTML">...</div>
```

```
←!— Replace entire element →  
<div hx-swap="outerHTML">...</div>
```

```
←!— Insert at beginning →  
<div hx-swap="afterbegin">...</div>
```

```
←!— Insert at end →  
<div hx-swap="beforeend">...</div>
```

Advanced Triggers

```
1 1!— Trigger on page load —>
2 <div hx-get="/data" hx-trigger="load">
3
4 4!— Trigger on intersection (lazy loading) —>
5 <div hx-get="/more" hx-trigger="intersect once">
6
7 7!— Debounced input —>
8 <input hx-get="/search"
9   hx-trigger="keyup changed delay:500ms">
10
11 11!— Multiple triggers —>
12 <div hx-get="/refresh"
13   hx-trigger="click, every 30s">
```

Advanced Triggers

```
1 1!— Trigger on page load —>
2 <div hx-get="/data" hx-trigger="load">
3
4 4!— Trigger on intersection (lazy loading) —>
5 <div hx-get="/more" hx-trigger="intersect once">
6
7 7!— Debounced input —>
8 <input hx-get="/search"
9   hx-trigger="keyup changed delay:500ms">
10
11 11!— Multiple triggers —>
12 <div hx-get="/refresh"
13   hx-trigger="click, every 30s">
```

Advanced Triggers

```
1 <!-- Trigger on page load -->
2 <div hx-get="/data" hx-trigger="load">
3
4 <!-- Trigger on intersection (lazy loading) -->
5 <div hx-get="/more" hx-trigger="intersect once">
6
7 <!-- Debounced input -->
8 <input hx-get="/search"
   hx-trigger="keyup changed delay:500ms">
10
11 <!-- Multiple triggers -->
12 <div hx-get="/refresh"
13   hx-trigger="click, every 30s">
```

Advanced Triggers

```
1 1. Trigger on page load →  
2 <div hx-get="/data" hx-trigger="load">  
3  
4 2. Trigger on intersection (lazy loading) →  
5 <div hx-get="/more" hx-trigger="intersect once">  
6  
7 3. Debounced input →  
8 <input hx-get="/search"  
9     hx-trigger="keyup changed delay:500ms">  
10  
11 4. Multiple triggers →  
12 <div hx-get="/refresh"  
13     hx-trigger="click, every 30s">
```

Loading Indicators

```
1 <button hx-post="/submit"
2           hx-indicator=".loading">
3   <span class="htmx-show">Submit</span>
4   <span class="loading htmx-indicator">
5     Submitting...
6   </span>
7 </button>
```

Loading Indicators

```
1 <button hx-post="/submit"
2           hx-indicator=".loading">
3   <span class="htmx-show">Submit</span>
4   <span class="loading htmx-indicator">
5     Submitting...
6   </span>
7 </button>
```

```
<script src="https://unpkg.com/htmrx.org@2.0.2"></script>
<script
  src="https://unpkg.com/htmrx.org/dist/ext/json-enc.js">
</script>
```

HTMX

```
1 <form
2     class="space-y-4"
3     hx-post="/waitlist"
4     hx-target="#container"
5     hx-swap="innerHTML"
6     hx-ext="json-enc"
7   >
8     <label class="w-full input validator">
9       <i class="h-6 hgi hgi-solid hgi-tick-02"></i>
10      <input
11          type="email"
12          name="email"
13          placeholder="hello@example.com"
```

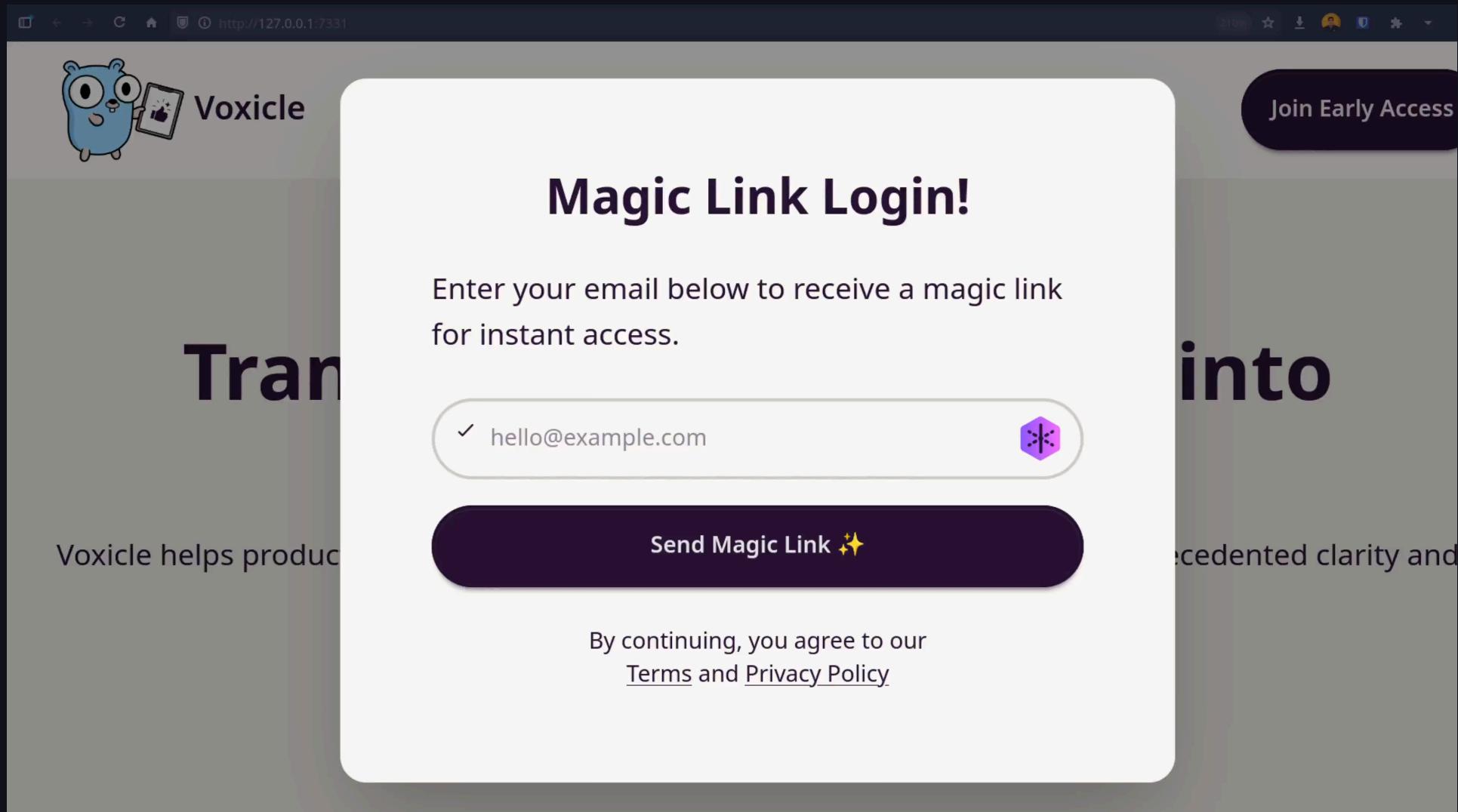
HTMX

```
0      fix-exe- json-enc
7  >
8    <label class="w-full input validator">
9      <i class="h-6 hgi hgi-solid hgi-tick-02"></i>
10     <input
11       type="email"
12       name="email"
13       placeholder="hello@example.com"
14       required
15     />
16   </label>
17   <div class="hidden validator-hint">
18     Enter valid email address
19   </div>
```

```
type Waitlist struct {
    Email string `json:"email"`
}
```

```
1 <div class="p-8 space-y-6 text-center">
2     <div class="flex justify-center text-neutral">
3         <i class="h-10 text-neutral hgi hgi-solid hgi-"
4     </div>
5     <h3 class="text-2xl font-semibold">
6         You're on the Waitlist 🎉
7     </h3>
8     <div class="space-y-6">
9         <p>Thank you for your interest in our applicat
10        <p>
11            We'll notify you at
12            <br />
13            <span class="font-mono text-primary">
```

```
8   <div class="space-y-6">
9     <p>Thank you for your interest in our applicat
10    <p>
11      We'll notify you at
12      <br />
13      <span class="font-mono text-primary">
14        hello@haseebmajid.dev
15      </span>
16      <br />
17      when we're ready to launch.
18    </p>
19  </div>
20</div>
```



A screenshot of a web browser showing the Voxicle Magic Link Login page. The page features a large blue cartoon character icon on the left and a "Join Early Access" button on the right. The main heading is "Magic Link Login!". Below it, a text instruction reads: "Enter your email below to receive a magic link for instant access." A text input field contains the email "hello@example.com" with a checkmark icon to its left. To the right of the input field is a purple hexagonal icon with a white asterisk symbol. A large dark blue button at the bottom is labeled "Send Magic Link ✨". Below the button, a note states: "By continuing, you agree to our [Terms](#) and [Privacy Policy](#)". The background of the page is a blurred version of the Voxicle interface.

Voxicle

Join Early Access

Magic Link Login!

Enter your email below to receive a magic link for instant access.

✓ hello@example.com

Send Magic Link ✨

By continuing, you agree to our
[Terms](#) and [Privacy Policy](#)

Why HTMX?

- State on backend
- Reduced complexity
- Simpler tooling

What about JSON?

- Separate API
- Mobile vs WebApp

React, vue, angular,
svelte...

HTMX



```
w.Header().Set("HX-Retarget", "#error_modal_container")  
w.Header().Set("Content-Type", "text/html")
```

WebSockets

```
1 <div
2   hx-ext="ws"
3   ws-connect="/ws">
4
5   <form
6     hx-vals='{"message_type": "submit_vote" }'
7     ws-send
8   >
9     <input name="voted_player_nickname" />
10    </form>
11 </div>
```

WebSockets

```
1 <div
2     hx-ext="ws"
3     ws-connect="/ws">
4
5     <form
6         hx-vals='{"message_type": "submit_vote" }'
7         ws-send
8     >
9         <input name="voted_player_nickname" />
10    </form>
11 </div>
```

WebSockets

```
1 <div
2   hx-ext="ws"
3   ws-connect="/ws">
4
5   <form
6     hx-vals='{"message_type": "submit_vote" }'
7     ws-send
8   >
9     <input name="voted_player_nickname" />
10    </form>
11 </div>
```

```
{  
  "message_type": "submit_vote",  
  "voted_player_nickname": "majiy"  
}
```

Caching Strategies

```
←!— Cache GET requests →  
<div hx-get="/expensive-data"  
     hx-trigger="load"  
     hx-cache="true">  
</div>  
  
←!— Conditional requests →  
<div hx-get="/data"  
     hx-headers='{"If-None-Match": "etag123"}'>  
</div>
```

HTMX Response Codes

- 204 - No Content • 304 - Not Modified • 4xx - Client errors • 5xx - Server errors

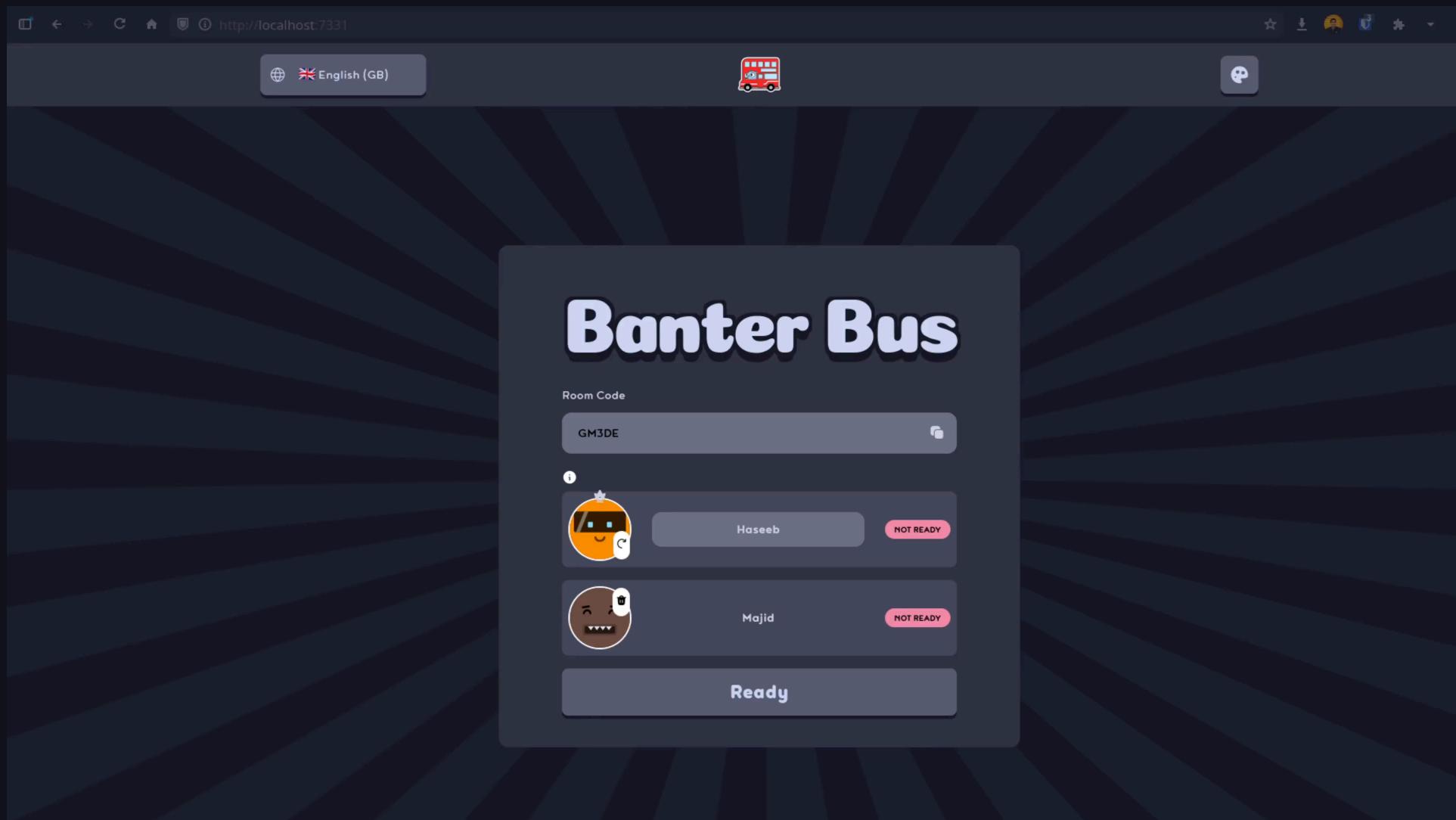
AlpineJS

```
<script  
src="https://cdn./.../alpinejs@3.14.3/dist/cdn.min.js">  
</script>
```

```
1 <div
2     x-data={ "showModal": false }
3     @keydown.escape="showModal = false"
4 >
5     <button type="button" @click="showModal = true">
6         <i class="hgi-information-circle"></i>
7     </button>
8     <div x-show="showModal">modal</div>
9 </div>
```

```
1 <div
2     x-data={ "showModal": false }
3     @keydown.escape="showModal = false"
4 >
5     <button type="button" @click="showModal = true">
6         <i class="hgi-information-circle"></i>
7     </button>
8     <div x-show="showModal">modal</div>
9 </div>
```

```
1 <div
2     x-data={ "showModal": false }
3     @keydown.escape="showModal = false"
4 >
5     <button type="button" @click="showModal = true">
6         <i class="hgi-information-circle"></i>
7     </button>
8     <div x-show="showModal">modal</div>
9 </div>
```



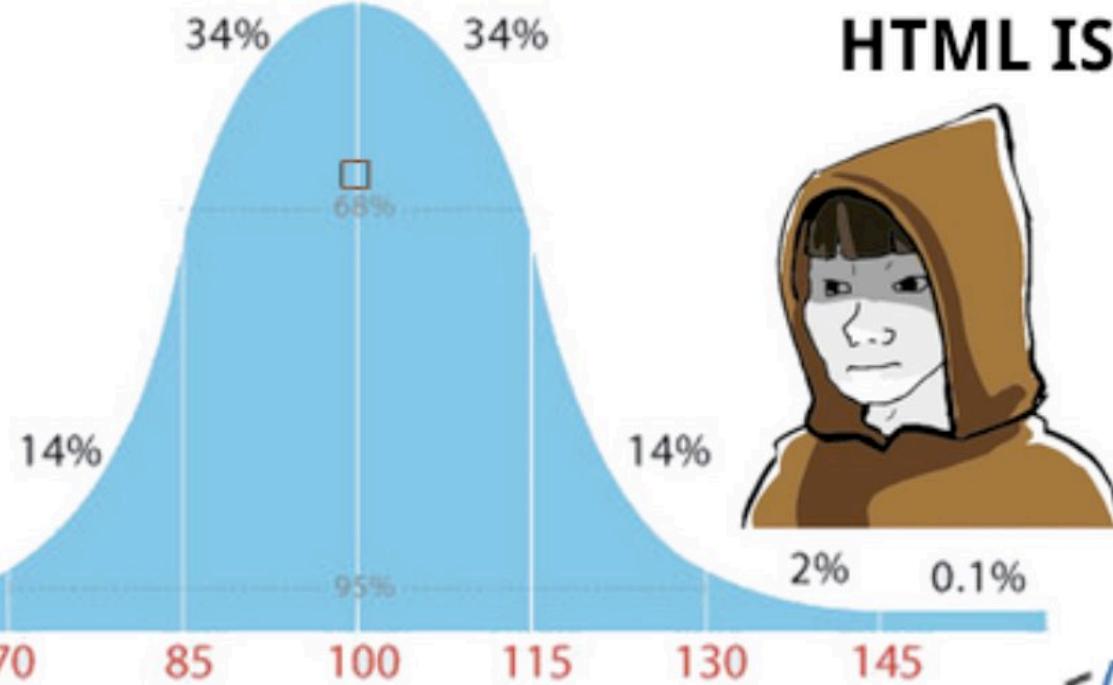
Alternatives

- Datastar
- Alpine AJAX

HTML IS FINE



0.1%
2%



wE hAvE tO uSe ReAcT &
ReAcT RoUtEr & GrApHqL
& rEdUx &...

HTML IS FINE



</> htmx

Backend

Handler

```
1 type Waitlist struct {
2     Email string `json:"email"`
3 }
4
5 func (h *Handler) AddToWaitlist(
6     w http.ResponseWriter,
7     r *http.Request,
8 ) {
9     var req Waitlist
10    body, _ := io.ReadAll(r.Body)
11    json.Unmarshal(body, &req)
12
13    waitlist, err := h.service.AddToWaitlist(
```

Handler

```
1 type Waitlist struct {
2     Email string `json:"email"`
3 }
4
5 func (h *Handler) AddToWaitlist(
6     w http.ResponseWriter,
7     r *http.Request,
8 ) {
9     var req Waitlist
10    body, _ := io.ReadAll(r.Body)
11    json.Unmarshal(body, &req)
12
13    waitlist, err := h.service.AddToWaitlist(
```

Handler

```
8  ) {
9     var req Waitlist
10    body, _ := io.ReadAll(r.Body)
11    json.Unmarshal(body, &req)
12
13    waitlist, err := h.service.AddToWaitlist(
14        r.Context(),
15        req.Email,
16    )
17    if err != nil {
18        http.Error(w,
19                    err.Error(),
20                    http.StatusInternalServerError,
21    }
```

Handler

```
15         req.Email,
16     )
17     if err != nil {
18         http.Error(w,
19             err.Error(),
20             http.StatusInternalServerError,
21         )
22     return
23 }
24
25 components.Waitlist(waitlist.Email).
26     Render(r.Context(), w)
27 }
```

Templ

- HTML Templates
- LSP
- Components

NORMAL 1 banterbus 2 Tab #2

banter

```
internal > views > sections > winner.templ > WinnerState
7 package sections
6
5 import (
4     "gitlab.com/hmajid2301/banterbus/internal/service"
3     "gitlab.com/hmajid2301/banterbus/internal/views/components"
2 )
1
8 templ Winner(state service.WinnerState, maxScore int) {
1    <div hx-swap-oob="innerHTML:#page">
2        <div class="mx-auto w-full max-w-5xl">
3            <div class="flex flex-col justify-center items-center space-y-4 text-text2">
4                <div class="grid grid-cols-1 gap-8 w-full lg:grid-cols-2 lg:gap-16">
5                    <div class="flex justify-center items-center">
6                        <div class="text-center">
7                            <h2 class="mb-4 text-2xl sm:text-3xl md:text-4xl">The winner is</h2>
8                            <p class="text-xl font-bold sm:text-2xl md:text-3xl text-gold">{ state.Players[0].Nickname }</p>
9
10                   </div>
11
12                   <div class="mx-auto w-full max-w-2xl">
13                       @components.Scoreboard(state.Players, maxScore)
14                   </div>
15
16               </div>
17           </div>
18 }
```

NORMAL <winner.templ > preview-env

html templ tailwindcss 1 8:28

```
1 package sections
2
3 import (
4     "gitlab.com/hmajid2301/banterbus/internal/service"
5     "gitlab.com/.../internal/views(blocks"
6 )
7
8 templ Winner(state service.WinnerState, maxScore int)
9 <div hx-swap-oob="innerHTML:#page">
10    <div>
11        <div class="flex">
12            <div class="grid">
13                <div>
```

```
1 package sections
2
3 import (
4     "gitlab.com/hmajid2301/banterbus/internal/service"
5     "gitlab.com/.../internal/views(blocks"
6 )
7
8 templ Winner(state service.WinnerState, maxScore int)
9 <div hx-swap-oob="innerHTML:#page">
10    <div>
11        <div class="flex">
12            <div class="grid">
13                <div>
```

```
2
3 import (
4     "gitlab.com/hmajid2301/banterbus/internal/service"
5     "gitlab.com/.../internal/views(blocks"
6 )
7
8 templ Winner(state service.WinnerState, maxScore int)
9 <div hx-swap-oob="innerHTML:#page">
10    <div>
11        <div class="flex">
12            <div class="grid">
13                <div>
14                    The winner is
```

```
4     gitlab.com/midjourney/dancerds/internal/service
5     "gitlab.com/.../internal/views/blocks"
6 )
7
8 templ Winner(state service.WinnerState, maxScore int)
9 <div hx-swap-oob="innerHTML:#page">
10    <div>
11        <div class="flex">
12            <div class="grid">
13                <div>
14                    The winner is
15                    { state.WinnerPlayer.Nickname }
16                </div>
17                @blocks Scoreboard()
```

```
8    import { withState } from '@dcl/react-service';
9    <div hx-swap-oob="innerHTML:#page">
10      <div>
11        <div class="flex">
12          <div class="grid">
13            <div>
14              The winner is
15              { state.WinnerPlayer.Nickname }
16            </div>
17            @blocks.Scoreboard(
18              state.Players,
19              maxScore,
20              )
21        </div>
```

```
12 <div class="status">
13     <div>
14         The winner is
15         { state.WinnerPlayer.Nickname }
16     </div>
17     @blocks.Scoreboard(
18         state.Players,
19         maxScore,
20         )
21     </div>
22     </div>
23 </div>
24 </div>
25 }
```

scripts.templ

```
1 templ Scripts(environment string) {
2 <script src="https://unpkg.com/htm.x.org@2.0.2">
3 </script>
4 <script src=".../dist/ext/json-enc.js">
5 </script>
6 <script src=".../alpinejs@3.14.3/dist/cdn.min.js">
7 </script>
8 @sentryLoad(environment)
9 }
```

scripts.templ

```
1 templ Scripts(environment string) {
2 <script src="https://unpkg.com/htm.x.org@2.0.2">
3 </script>
4 <script src=".../dist/ext/json-enc.js">
5 </script>
6 <script src=".../alpinejs@3.14.3/dist/cdn.min.js">
7 </script>
8 @sentryLoad(environment)
9 }
```

```
1 script sentryLoad(environment string) {
2   Sentry.onLoad(function() {
3     Sentry.init({
4       environment: environment,
5     });
6   });
7 }
```

```
1 script sentryLoad(environment string) {
2   Sentry.onLoad(function() {
3     Sentry.init({
4       environment: environment,
5     });
6   });
7 }
```

layout.templ

```
1 package layouts
2
3 import "gitlab.com/.../http/views/components"
4
5 templ Base(title string, environment string) {
6     <!DOCTYPE html>
7     <html lang="en">
8         <head>
9             </head>
10            <body class="bg-base-200 text-neutral">
11                { children... }
12            </body>
13        </html>
```

layout.templ

```
2
3 import "gitlab.com/.../http/views/components"
4
5 templ Base(title string, environment string) {
6     <!DOCTYPE html>
7     <html lang="en">
8         <head>
9             </head>
10            <body class="bg-base-200 text-neutral">
11                { children... }
12            </body>
13        </html>
14 }
```

layout.templ

```
2
3 import "gitlab.com/.../http/views/components"
4
5 templ Base(title string, environment string) {
6     <!DOCTYPE html>
7     <html lang="en">
8         <head>
9             </head>
10            <body class="bg-base-200 text-neutral">
11                { children... }
12            </body>
13        </html>
14 }
```

```
1 templ Dashboard(title string, environment string) {
2   @Base(title, environment) {
3     <div class="drawer lg:drawer-open">
4       </div>
5     }
6 }
```



<http://localhost:7331>

Transform user feedback into product gold

Voxicle helps product teams collect, prioritize, and act on user insights with unprecedented clarity and speed.

[Join Early Access](#) [See Features](#)

i18n

```
1 func (m Middleware) Locale(next http.Handler)
2 http.Handler {
3     return http.HandlerFunc(
4         func(w http.ResponseWriter, r *http.Request) {
5             locale := extractLocaleFromURL(r.URL.Path)
6             ctx, err := ctxi18n.WithLocale(
7                 r.Context(),
8                 locale,
9             )
10            next.ServeHTTP(w, r.WithContext(ctx)))
11        })
12 }
```

i18n

```
1 func (m Middleware) Locale(next http.Handler)
2 http.Handler {
3     return http.HandlerFunc(
4         func(w http.ResponseWriter, r *http.Request) {
5             locale := extractLocaleFromURL(r.URL.Path)
6             ctx, err := ctxi18n.WithLocale(
7                 r.Context(),
8                 locale,
9             )
10            next.ServeHTTP(w, r.WithContext(ctx))
11        })
12    }
```

i18n

```
1 func (m Middleware) Locale(next http.Handler)
2 http.Handler {
3     return http.HandlerFunc(
4         func(w http.ResponseWriter, r *http.Request) {
5             locale := extractLocaleFromURL(r.URL.Path)
6             ctx, err := ctxi18n.WithLocale(
7                 r.Context(),
8                 locale,
9             )
10            next.ServeHTTP(w, r.WithContext(ctx)))
11        })
12 }
```

i18n

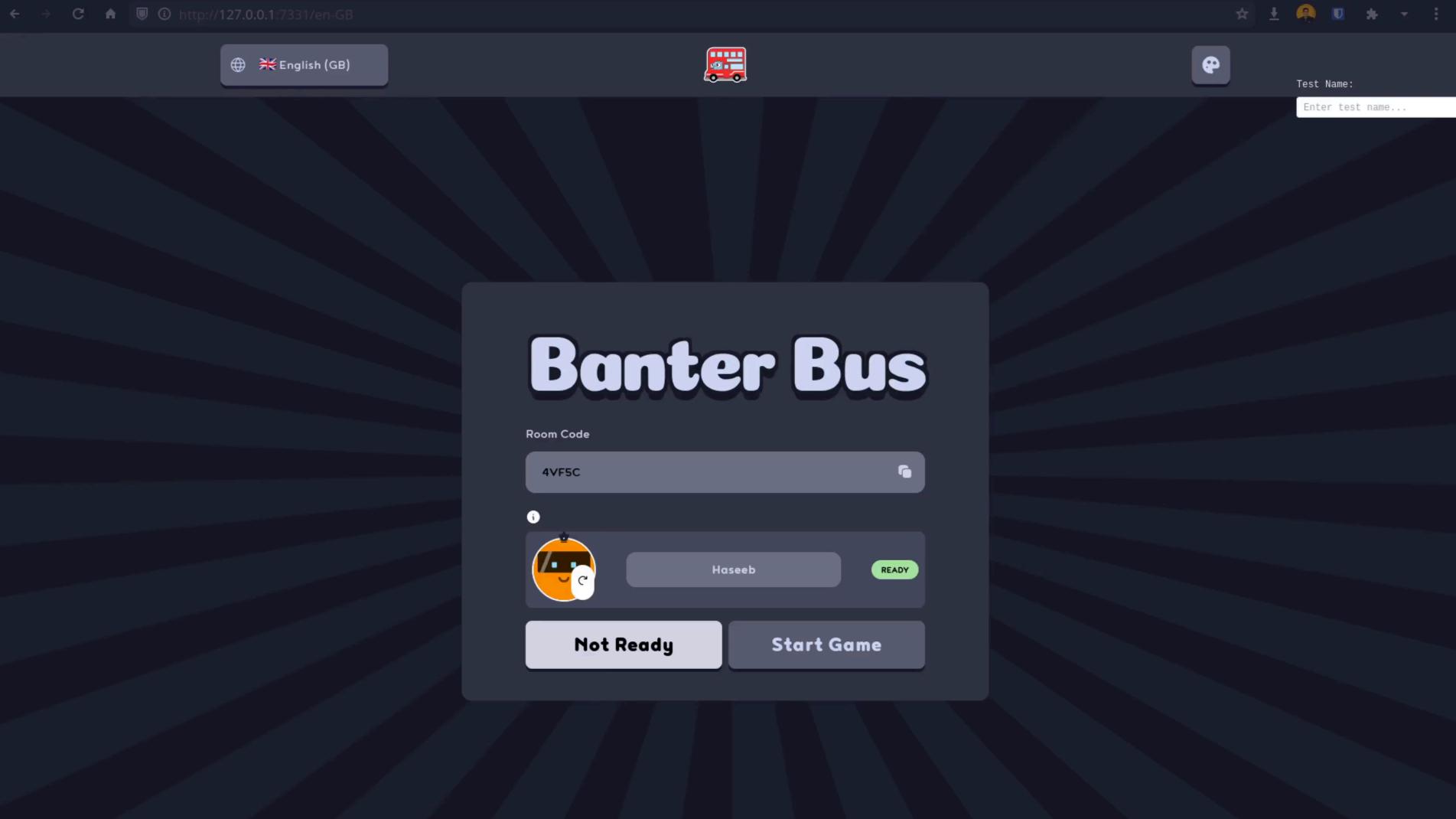
```
1 func (m Middleware) Locale(next http.Handler)
2 http.Handler {
3     return http.HandlerFunc(
4         func(w http.ResponseWriter, r *http.Request) {
5             locale := extractLocaleFromURL(r.URL.Path)
6             ctx, err := ctxi18n.WithLocale(
7                 r.Context(),
8                 locale,
9             )
10            next.ServeHTTP(w, r.WithContext(ctx))
11        })
12 }
```

```
1 component := sections.Winner(winnerState, maxScore)
2 err := component.Render(r.Context(), &buf)
3 if err != nil {
4     return err
5 }
```

```
en-GB:  
common:  
    ready_button: "Ready"  
    roomcode_label: "Room Code"  
home:  
    start_button_label: "Start Game"
```

```
<div>
  { i18n.T(ctx, "common.ready_button") }
</div>
```

```
<div class="...>
    Ready
</div>
```



Postgres

sqlc

```
1 version: "2"
2 sql:
3   - engine: "postgresql"
4     queries: "internal/store/db/sqlc/query.sql"
5     schema: "internal/store/db/sqlc/migrations"
6   gen:
7     go:
8       package: "db"
9       out: "internal/store/db"
10      sql_package: "pgx/v5"
11      emit_interface: true
```

sqlc

```
1 version: "2"
2 sql:
3   - engine: "postgresql"
4     queries: "internal/store/db/sqlc/query.sql"
5     schema: "internal/store/db/sqlc/migrations"
6   gen:
7     go:
8       package: "db"
9       out: "internal/store/db"
10      sql_package: "pgx/v5"
11      emit_interface: true
```

sqlc

```
1 version: "2"
2 sql:
3   - engine: "postgresql"
4     queries: "internal/store/db/sqlc/query.sql"
5     schema: "internal/store/db/sqlc/migrations"
6   gen:
7     go:
8       package: "db"
9       out: "internal/store/db"
10      sql_package: "pgx/v5"
11      emit_interface: true
```

query.sql

```
-- name: AddUser :one
insert into users (email) values ($1) returning *;
```

```
sqlc generate
```

generated

```
1 const addUser = `-- name: AddUser :one
2 insert into users (email) values ($1) returning id, cr
3 `
4
5 func (q *Queries) AddUser(
6     ctx context.Context,
7     email string,
8 ) (User, error) {
9     row := q.db.QueryRow(ctx, addUser, email)
10    var i User
11    err := row.Scan(
12        &i.ID,
13        &i.CreatedAt,
```

generated

```
1 const addUser = `-- name: AddUser :one
2 insert into users (email) values ($1) returning id, cr
3 `
4
5 func (q *Queries) AddUser(
6     ctx context.Context,
7     email string,
8 ) (User, error) {
9     row := q.db.QueryRow(ctx, addUser, email)
10    var i User
11    err := row.Scan(
12        &i.ID,
13        &i.CreatedAt,
```

generated

```
6     ctx context.Context,
7     email string,
8 ) (User, error) {
9     row := q.db.QueryRow(ctx, addUser, email)
10    var i User
11    err := row.Scan(
12        &i.ID,
13        &i.CreatedAt,
14        &iUpdatedAt,
15        &i.Email,
16    )
17    return i, err
18 }
```

```
type Querier interface {
    AddFibbingItRole(ctx context.Context, arg AddFibbingItRoleParams) (Room, error)
    AddPlayer(ctx context.Context, arg AddPlayerParams) (Player, error)
    AddQuestion(ctx context.Context, arg AddQuestionParams) (Question, error)
    AddQuestionTranslation(ctx context.Context, arg AddQuestionTranslationParams) (Question, error)
    AddRoom(ctx context.Context, arg AddRoomParams) (Room, error)
    GetAllPlayerByRoomCode(ctx context.Context, roomCode string) ([]Player, error)
    // ...
}
```

Goose

```
1 -- +goose Up
2 -- +goose StatementBegin
3 CREATE TABLE IF NOT EXISTS feedback (
4     id UUID PRIMARY KEY DEFAULT generate_uuidv7(),
5     created_at TIMESTAMP DEFAULT current_timestamp,
6     updated_at TIMESTAMP DEFAULT current_timestamp,
7     title TEXT NOT NULL,
8     description TEXT NOT NULL,
9 );
10 -- +goose StatementEnd
11
12 -- +goose Down
13 -- +goose StatementBegin
```

Goose

```
3 CREATE TABLE IF NOT EXISTS feedback (
4     id UUID PRIMARY KEY DEFAULT generate_uuidv7(),
5     created_at TIMESTAMP DEFAULT current_timestamp,
6     updated_at TIMESTAMP DEFAULT current_timestamp,
7     title TEXT NOT NULL,
8     description TEXT NOT NULL,
9 );
10 -- +goose StatementEnd
11
12 -- +goose Down
13 -- +goose StatementBegin
14 DROP TABLE IF EXISTS feedback;
15 -- +goose StatementEnd
```

Transactions

```
1 func (s *DB) StartGame(ctx context.Context, arg StartGameArg) (*Game, error) {
2     return s.Transaction(ctx, func(q *Queries) error {
3         err := q.AddGame(arg)
4         if err != nil {
5             return err
6         }
7         _, err := q.UpdateRoomState(ctx, UpdateParams{
8             RoomState: Playing.String(),
9             ID:        arg.RoomID,
10            })
11        if err != nil {
12            return err
13        }
14        return q.GetGame(arg.RoomID)
15    })
16}
```

Transactions

```
1 func (s *DB) StartGame(ctx context.Context, arg StartGameArg) (*Game, error) {
2     return s.Transaction(ctx, func(q *Queries) error {
3         err := q.UpdateRoomState(ctx, UpdateParams{
4             RoomState: Playing.String(),
5             ID:        arg.RoomID,
6         })
7         if err != nil {
8             return err
9         }
10    })
11 }
12
13 // Add game state
```

Transactions

```
9 if err != nil {
10     return err
11 }
12
13 // Add game state
14 _, err = q.AddGameState(ctx, AddGameStateParams{
15     ID:        arg.GameStateID,
16     RoomID:    arg.RoomID,
17     State:    FibbingITQuestion.String(),
18 })
19 if err != nil {
20     return err
21 }
22
```

```
1 type Storer interface {
2     db.Querier
3     StartGame(ctx context.Context, arg db.StartGameArgs) (db.Game, error)
4 }
```

```
1 type Storer interface {
2     db.Querier
3     StartGame(ctx context.Context, arg db.StartGameArgs) (db.Game, error)
4 }
```

```
1 type Storer interface {
2     db.Querier
3     StartGame(ctx context.Context, arg db.StartGameArgs) (db.Game, error)
4 }
```

DevEx

docker-compose.yml

```
services:  
  postgres:  
    image: postgres:17.4  
    ports:  
      - "5432:5432"  
    environment:  
      POSTGRES_USER: postgres  
      POSTGRES_PASSWORD: postgres  
    volumes:  
      - postgres-data:/var/lib/postgresql/data  
      - ./init.sql:/docker-entrypoint-initdb.d/init.sql
```

Taskfile.yml

```
1 version: "3"
2
3 tasks:
4   build:
5     desc: Build the binary in a tmp location.
6     cmd:
7       - go build -o ./tmp/main ./cmd/server/main.go
8
9   dev:
10    desc: Start the app in dev mode with live-reloading.
11    dotenv:
12      - .env.local
13    cmd:
```

Taskfile.yml

```
4   build:
5     desc: Build the binary in a tmp location.
6     cmd:
7       - go build -o ./tmp/main ./cmd/server/main.go
8
9   dev:
10    desc: Start the app in dev mode with live-reloading
11    dotenv:
12      - .env.local
13    cmd:
14      - docker compose up -d
15      - task: watch
16      - air
```

Taskfile.yml

```
4   build:
5     desc: Build the binary in a tmp location.
6     cmd:
7       - go build -o ./tmp/main ./cmd/server/main.go
8
9   dev:
10    desc: Start the app in dev mode with live-reloading
11    dotenv:
12      - .env.local
13    cmd:
14      - docker compose up -d
15      - task: watch
16      - air
```

Taskfile.yml

```
4   build:
5     desc: Build the binary in a tmp location.
6     cmd:
7       - go build -o ./tmp/main ./cmd/server/main.go
8
9   dev:
10    desc: Start the app in dev mode with live-reloading
11    dotenv:
12      - .env.local
13    cmd:
14      - docker compose up -d
15      - task: watch
16      - air
```

.air.toml

```
1 [build]
2 bin = "./tmp/main"
3 cmd = "task build"
4 exclude_dir = ["assets", "tmp", "vendor", "testdata"]
5 include_ext = ["go", "css", "templ"]
6 exclude_regex = ["_test.go"]
```

.air.toml

```
1 [build]
2 bin = "./tmp/main"
3 cmd = "task build"
4 exclude_dir = ["assets", "tmp", "vendor", "testdata"]
5 include_ext = ["go", "css", "templ"]
6 exclude_regex = ["_test.go"]
```

```
watch:  
desc: Watch for file changes  
cmds:  
- templ generate -watch --open-browser=true &  
- tailwindcss --watch=always -i \  
    ./static/css/tailwind.css -o \  
    ./static/css/styles.css --minify &
```

Nix

```
example on main via 🐕 v1.22.8  
› which golangci-lint
```

```
example on main via 🐕 v1.22.8  
› nix develop
```

```
example on main via 🐕 v1.22.8 ❄️ impure (nix-shell-env)  
› which golangci-lint  
/nix/store/kcd...golangci-lint-1.56.2/bin/golangci-lint
```


When not to use HTMX?

- Lots of frontend reactivity
- Separate frontend/backend teams
- Design System

Other Issues?

- Alpine: Stringified JS
- Templ: Another tool
- SQLC: Dynamic queries

Further

- Observability
 - OTEL
- Playwright
 - Go



- <https://haseebmajid.dev/slides/go-labs-htmx-go-web-app/>
- Banter Bus:
<https://gitlab.com/hmajid2301/banterbus>

Useful Links

- Banter Bus: <https://gitlab.com/hmajid2301/banterbus>
- Nix Dev Shell: https://www.youtube.com/watch?v=bdGfn_ihHOk
- Playwright: <https://www.youtube.com/watch?v=JyfXWzqBjwU>
- OTel & Go: <https://www.youtube.com/watch?v=IrxNwk&list=PLSCmmmcxRB6DilKhSz09JL9F4CVI>