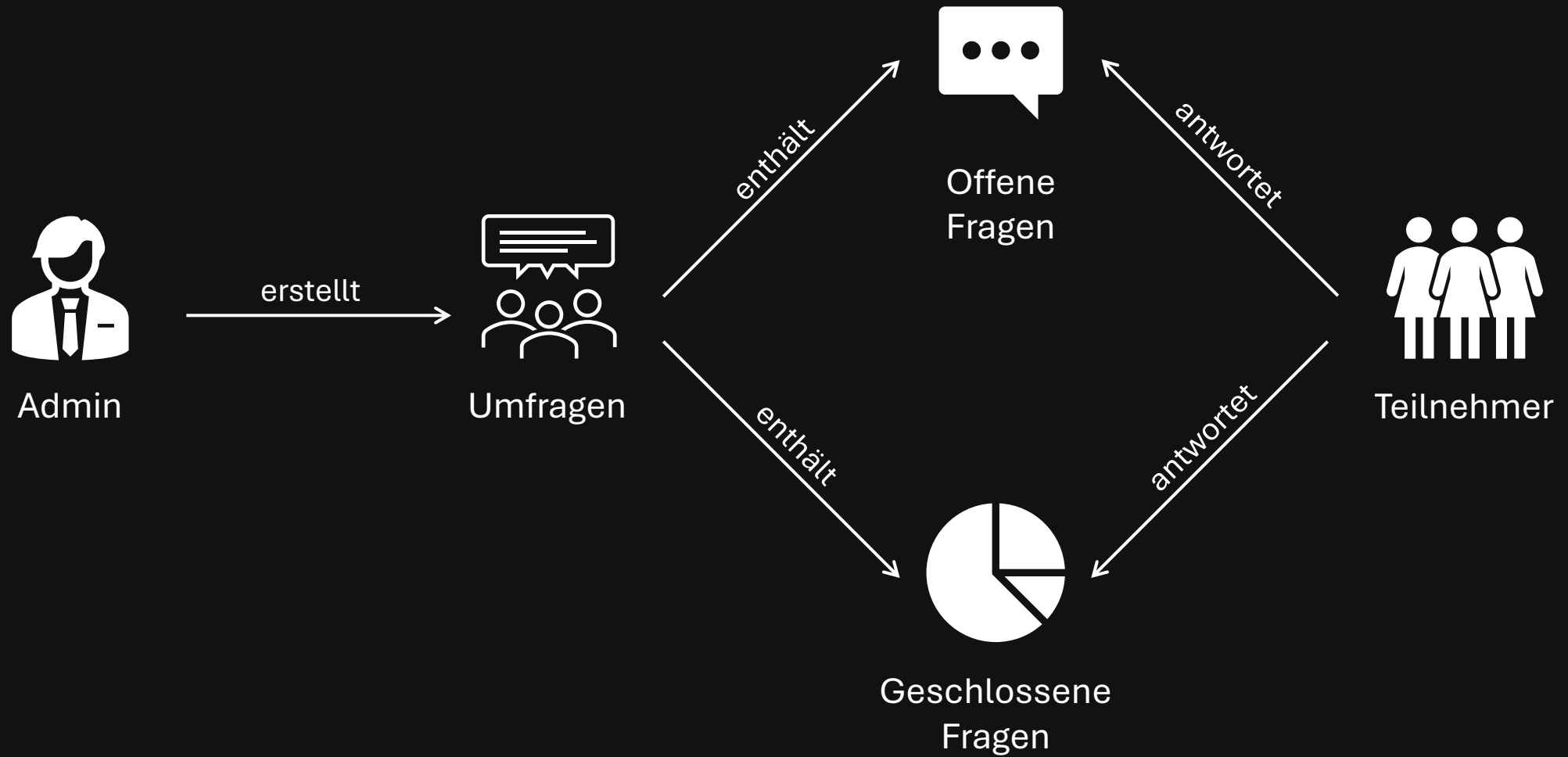


PollMeMaybe

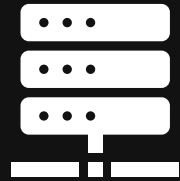
Umfragen leichtgemacht



IDEE



ANFORDERUNGEN



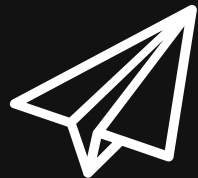
Load-
Balancing



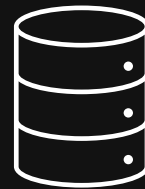
Dockerized



Einfaches
Frontend



Server-Sent
Events



Datenbank



Latest
Releases

LIVE-DEMO

Welcome to PollMeMaybe!

If you wish to create a poll, enter the name and click "create":

CREATE

If you have an admin or join code, you can enter it here:

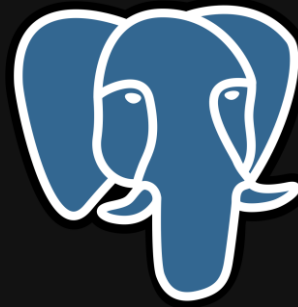
 

SUBMIT

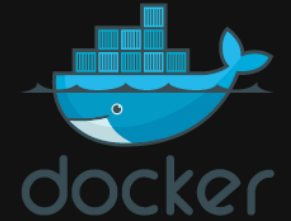
TECHNOLOGIEN



Vue.js



PostgreSQL



Docker

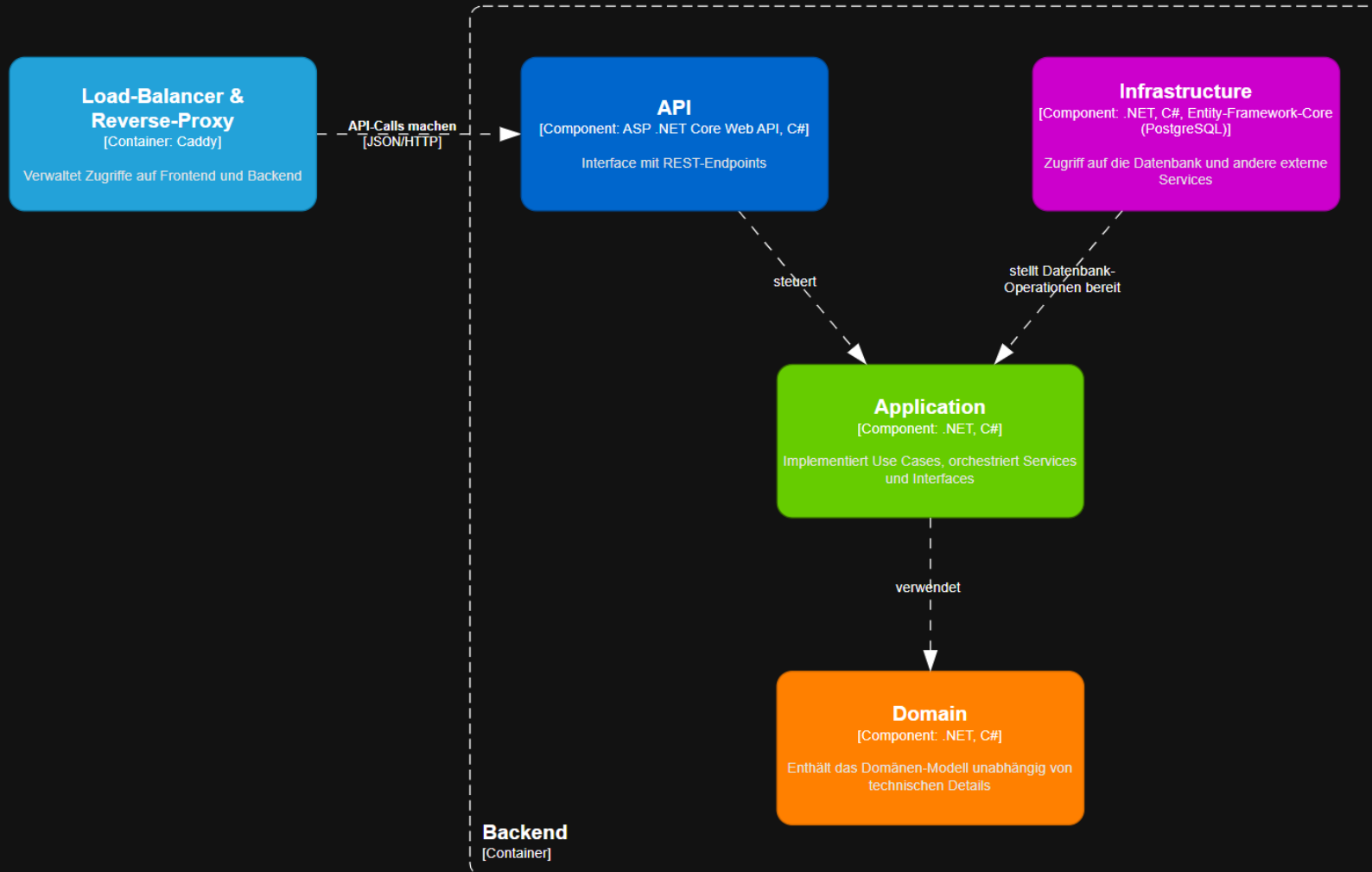


.NET 9.0

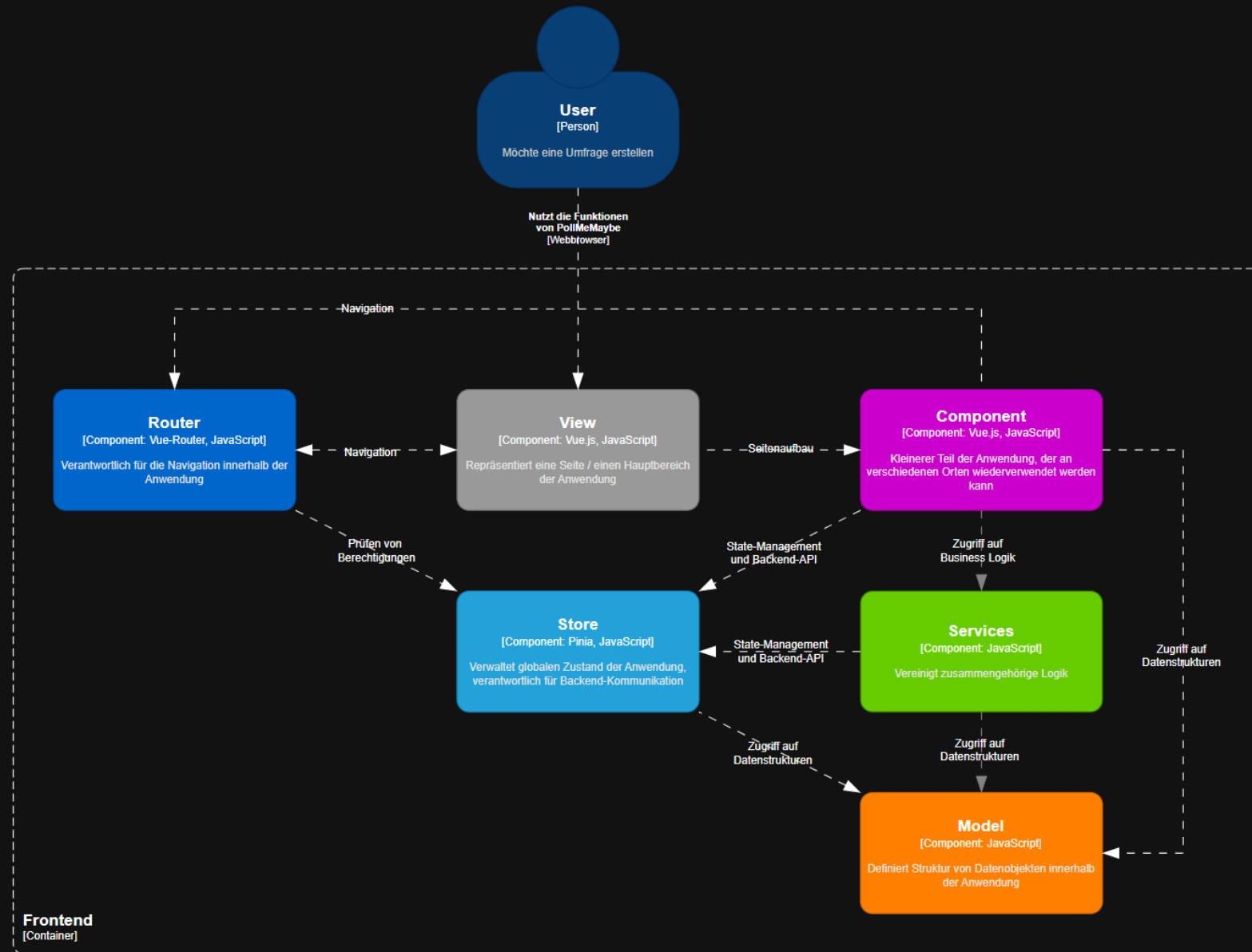


Caddy

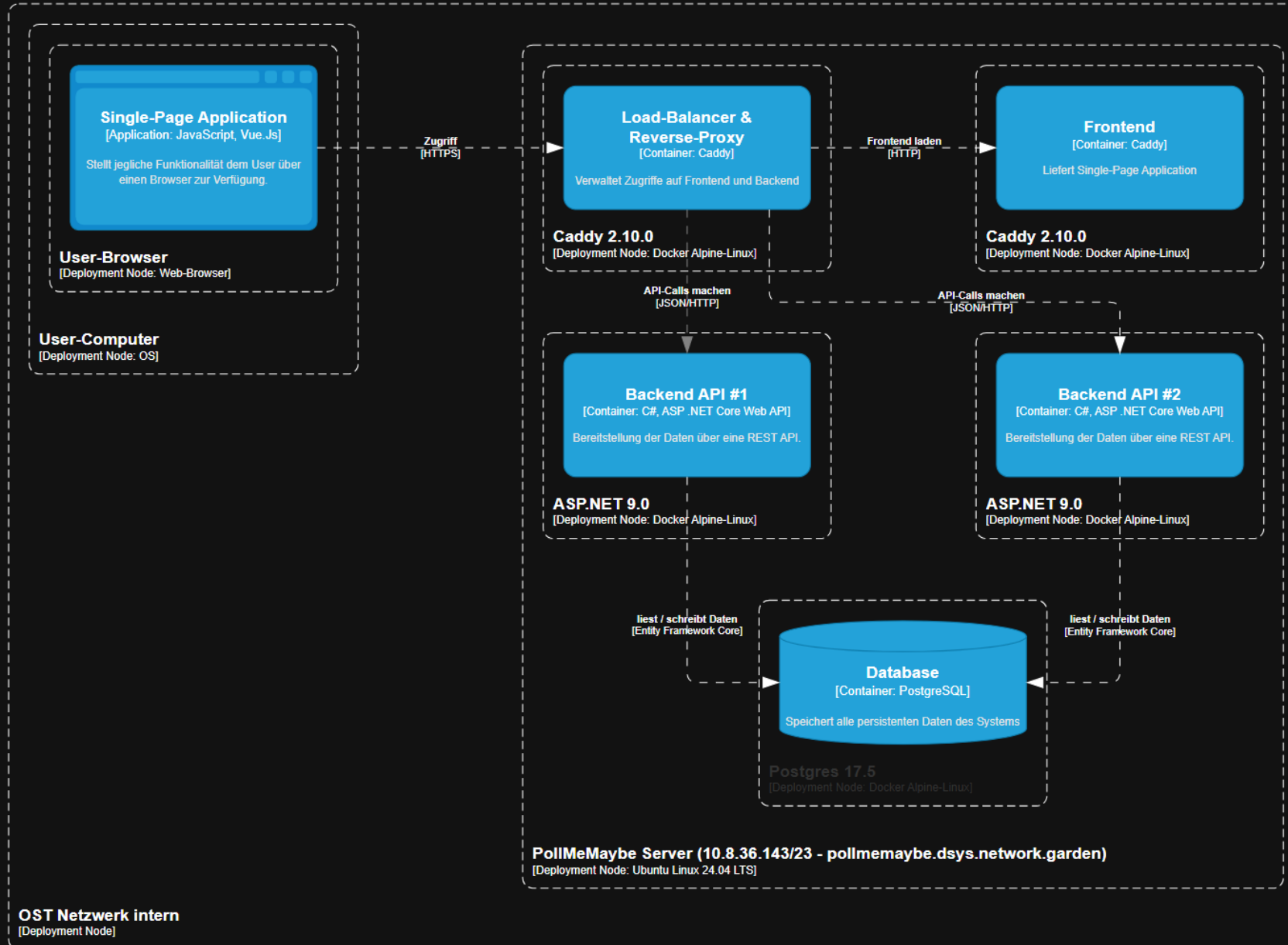
BACKEND-ARCHITEKTUR



FRONTEND-ARCHITEKTUR



DEPLOYMENT-VIEW

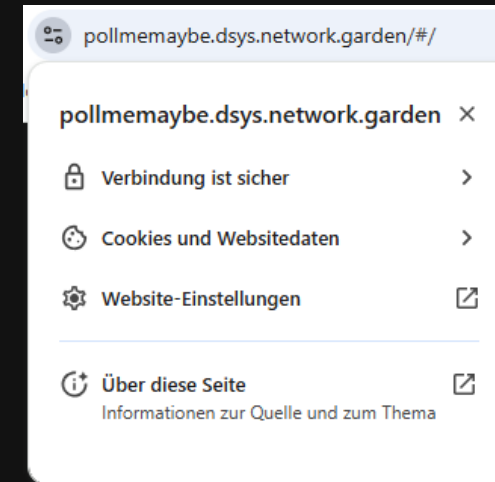


CADDY-KONFIGURATION

```
https://pollmemaybe.dsys.network.garden {  
  tls {  
    propagation_timeout -1  
    dns acmedns {  
      username {$ACME_DNS_USERNAME}  
      password {$ACME_DNS_PASSWORD}  
      subdomain {$ACME_DNS_SUBDOMAIN}  
      server_url https://acme-dns.ins.work:8844  
    }  
  }  
  
  handle /api/* {  
    reverse_proxy backend1:8080 backend2:8080 {  
      lb_policy random  
      health_uri /healthz  
      health_interval 5s  
    }  
  }  
  
  handle {  
    reverse_proxy frontend:80  
  }  
}
```

HTTPS

Reverse-Proxy &
Load-Balancer



SERVER-SENT EVENTS

Backend

```
[HttpGet("currentQuestion")]
0 references
public async Task GetCurrentQuestionUpdates([FromQuery] string joinCode)
{
    Response.Headers.Append("Content-Type", "text/event-stream");
    Response.Headers.Append("Cache-Control", "no-cache");
    Response.Headers.Append("Connection", "keep-alive");

    var cancellation_token = HttpContext.RequestAborted;

    while (!cancellation_token.IsCancellationRequested)
    {
        var poll = await _liveUpdateService.GetPoll(joinCode);

        // some code

        var json = JsonSerializer.Serialize(payload, options);
        await Response.WriteAsync($"data: {json}\n\n", cancellation_token);

        await Response.Body.FlushAsync(cancellation_token);

        await Task.Delay(500, cancellation_token);
    }
}
```

Frontend

```
function openSSE() {
    if (eventSource.value) eventSource.value.close()

    eventSource.value = new EventSource(sseUrl.value)

    eventSource.value.addEventListener('open', () => {
        lastMessageAt = Date.now()
    })

    eventSource.value.addEventListener('message', (event) => {
        lastMessageAt = Date.now()

        const parsed = JSON.parse(event.data)

        if (parsed.state === POLL_STATES.STARTED) {
            currentQuestion.value = null
            pollState.value = POLL_STATES.STARTED
        } else if (parsed.state === POLL_STATES.FINISHED) {
            pollState.value = POLL_STATES.FINISHED
            currentQuestion.value = null
            stopHealthCheck()
            closeSSE()
        } else if (parsed.state === POLL_STATES.RUNNING) {
            // some code
            currentQuestion.value = parsed
        }
    })

    eventSource.value.addEventListener('close', () => {
        closeSSE()
    })
}

function closeSSE() {
    if (eventSource.value) {
        eventSource.value.close()
        eventSource.value = null
    }
}

function startHealthCheck() {
    stopHealthCheck()
    healthCheckInterval = setInterval(() => {
        const now = Date.now()
        const elapsed = now - lastMessageAt
        if (elapsed > 1000) {
            closeSSE()
            openSSE()
            lastMessageAt = Date.now()
        }
    }, 1000)
}

function stopHealthCheck() {
    if (healthCheckInterval) {
        clearInterval(healthCheckInterval)
        healthCheckInterval = null
    }
}
```

Fazit

Schwierigkeiten



SSL-
Zertifikate



Performance

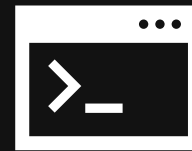
Positive Erfahrungen



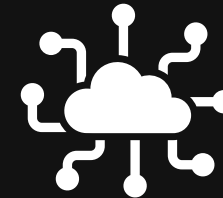
Sehr gute
Betreuung



Spannende
Aufgabe



Vorlage für
andere Projekte



Modul
übergreifend

Q&A

