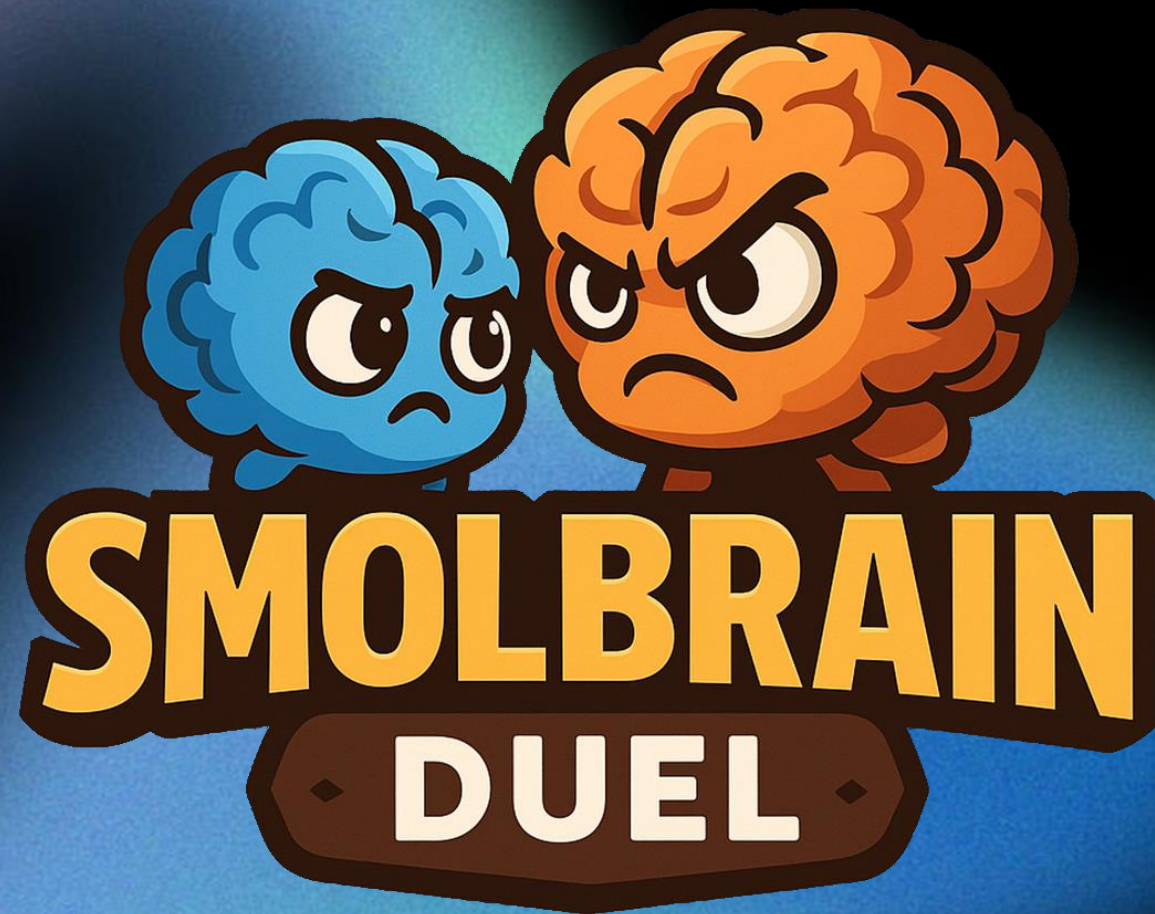






Motivation

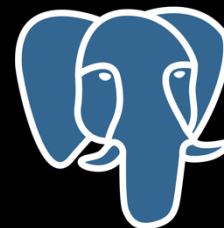
Inspiration von JKLM

Praktischer Nutzen des Projekts

Something fun!



Komponenten / Frameworks



A X 1 O S



Mono-Repo

- Ordner pro Service
- CI Pipeline – Linter

GROUP_QUIZ

> assets

> backend

> frontend

> nginx

> postgres

🔗 .gitignore

🐱 .gitlab-ci.yml

🚢 docker-compose.yml

📖 readme.md

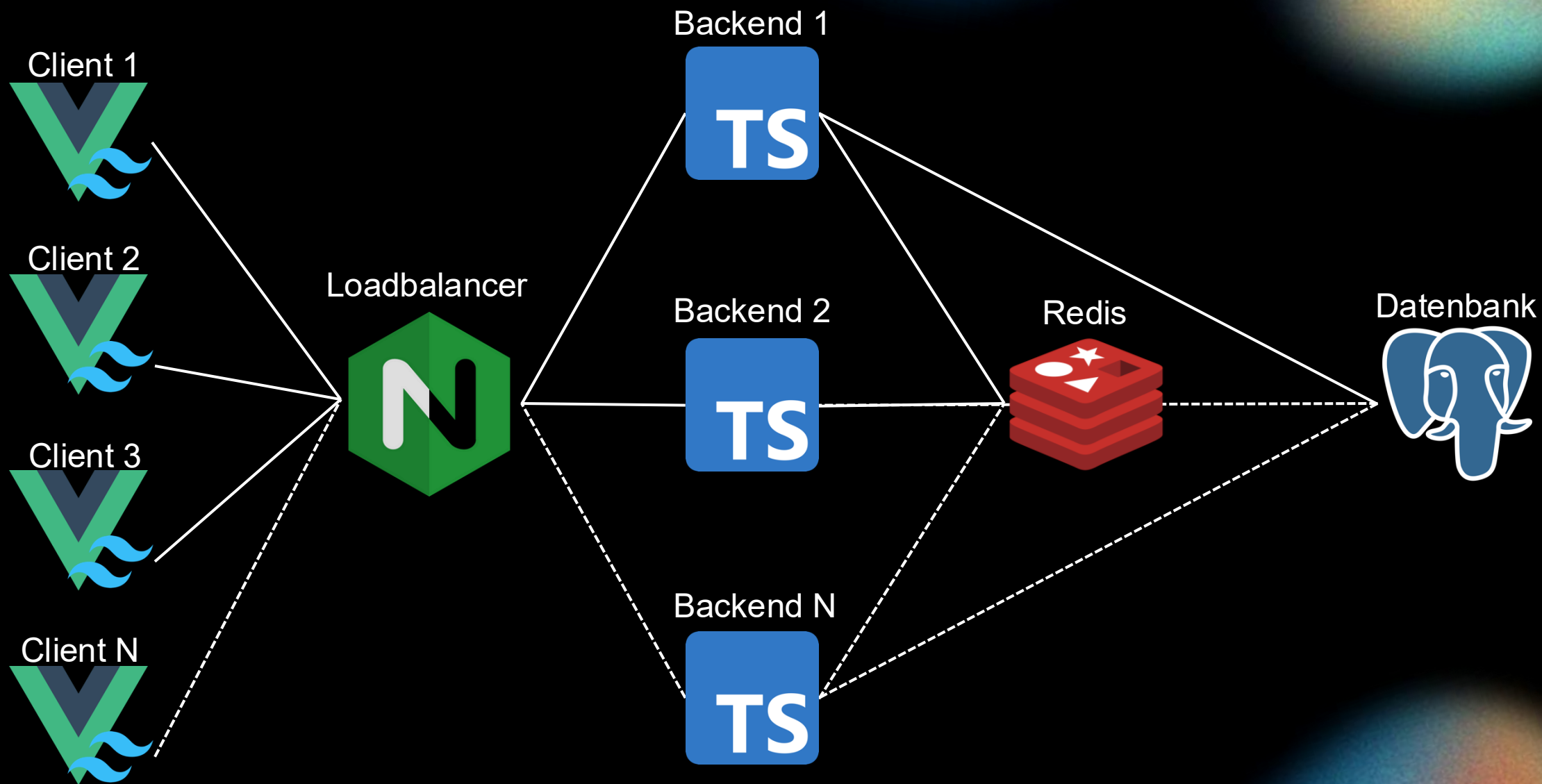


- `docker-compose up --build --scale backend=2 -d`
- Dockerfile pro service



Docker

☐	Name	Container ID	Image	Port(s)	CPU (%)	Last sta	Actions
☐	group_quiz	-	-	-	5.4%	12 minu	☐ ⋮
☐	redis-1	203d8f66e2df	redis:alpine	6379:6379	0.7%	12 minu	☐ ⋮
☐	database-1	014c6b07ae01	postgres:la	5432:5432	0%	12 minu	☐ ⋮
☐	backend-2	ef8140d0f627	group_quiz		1.66%	12 minu	☐ ⋮
☐	backend-1	a82025c31fb4	group_quiz		1.79%	12 minu	☐ ⋮
☐	frontend-1	93909f45025b	group_quiz	5173:5173	1.23%	12 minu	☐ ⋮
☐	loadbalancer	eb21e06a578d	nginx:lates	8080:80	0.02%	12 minu	☐ ⋮





Komponenten im Detail





Frontend

Vue.js

PlayerNameInput

Namesauswahl

Lobbies

Lobby Auswahl

Lobby

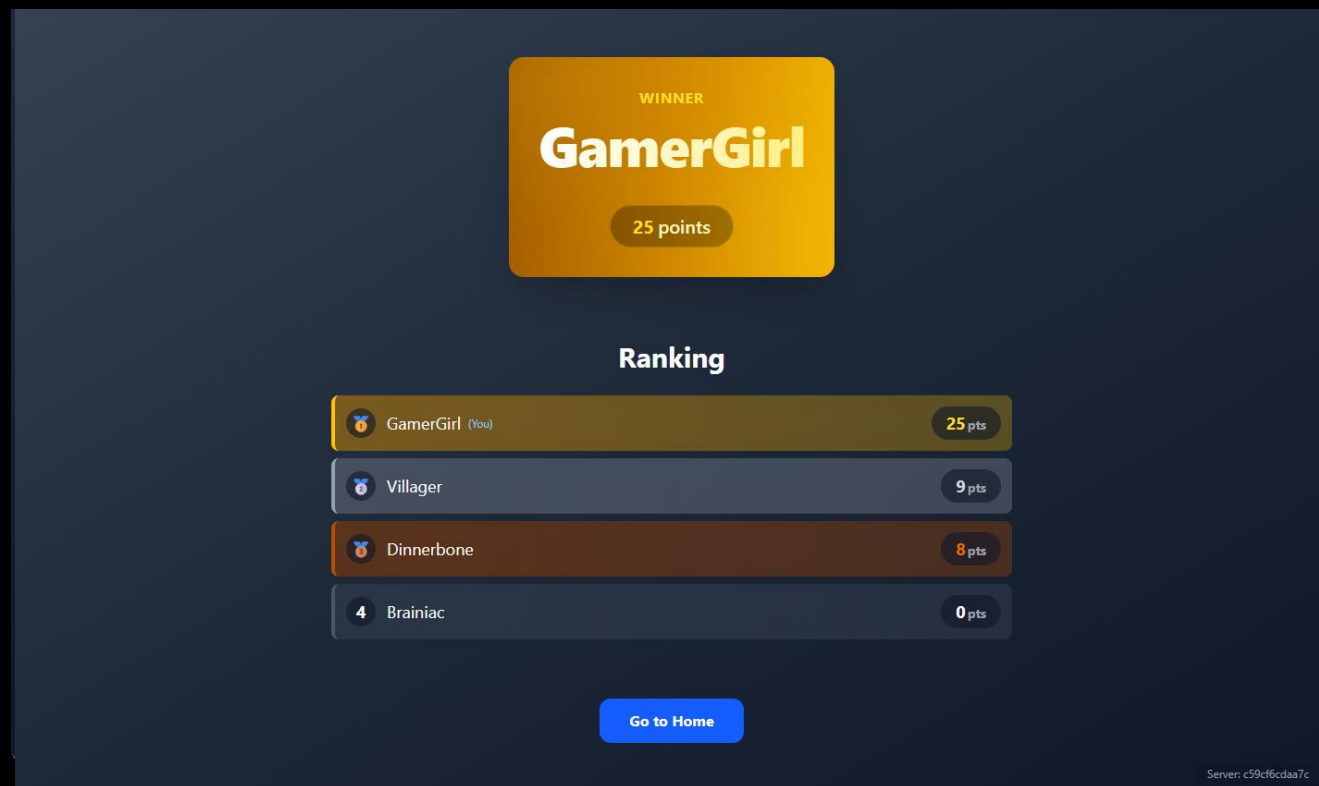
Lobby Konfiguration

Game

Quizfragen

Scoreboard

Punktliste





Backend

- Express JS
- Stateless
- Message-Broker via Redis
- Logging / Error-Handling

```
[2025-05-25T18:06:06.928Z] [INFO] [LobbyRoutes] Timer update event emitted for lobby 539Q
[2025-05-25T18:06:06.928Z] [INFO] [RedisService] Published message to channel lobby-events-539Q
[2025-05-25T18:06:06.928Z] [INFO] [RedisService] Received message from channel lobby-events-539Q
[2025-05-25T18:06:06.941Z] [INFO] [requestLogger] Incoming request: GET /server
[2025-05-25T18:06:07.667Z] [INFO] [requestLogger] Incoming request: POST /game/539Q/answer
[2025-05-25T18:06:07.674Z] [INFO] [GameService] Updated score for player 9 in lobby 539Q. Points added: 6.
[2025-05-25T18:06:07.677Z] [INFO] [GameService] State updated for lobby 539Q: {
  state: 'correct',
  timer: 15,
  currentState: 'waiting',
  totalQuestions: 10,
  currentQuestion: {
    id: 6,
    tag: 'athlete',
    answer: [ 'michael jordan', 'jordan' ],
    picture: 'assets/021.jpg',
    question: 'Who is this famous athlete?'
  },
  questionsRemaining: 4,
  currentQuestionIndex: 6
}
```



Loadbalancer

- Loadbalancing
- Reverse Proxy
- Dynamische Lastverteilung
- Config

```
upstream frontend {  
    server frontend:5173;  
}  
  
upstream backend {  
    zone upstream_dynamic 64k;  
    server group_quiz-backend-1:3000 resolve;  
    server group_quiz-backend-2:3000 resolve;  
}
```




Loadbalancer

- Loadbalancing
- Reverse Proxy
- Dynamische Lastverteilung
- Config

```
location /api/ {  
    rewrite ^/api(/.*)$ $1 break; # Removes '/api' prefix before forwarding  
    proxy_pass http://backend;  
  
    # Set headers for backend communication  
    proxy_set_header Host $host;  
    proxy_set_header X-Real-IP $remote_addr;  
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
    proxy_set_header X-Forwarded-Proto $scheme;  
  
    # Enable logging for the proxying requests  
    access_log /var/log/nginx/backend_access.log;  
}
```



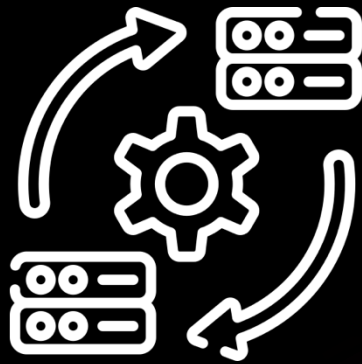
Loadbalancer

- Loadbalancing
- Reverse Proxy
- Dynamische Lastverteilung
- Config

```
location /events/ {  
    proxy_pass http://backend; # Ensure this matches your SSE endpoint  
    proxy_http_version 1.1;  
    proxy_set_header Upgrade $http_upgrade;  
    proxy_set_header Connection "keep-alive"; # Keep connection open  
    proxy_set_header Host $host;  
    proxy_set_header X-Real-IP $remote_addr;  
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
    proxy_set_header X-Forwarded-Proto $scheme;  
  
    # Disable buffering (important for SSE to work properly)  
    proxy_buffering off;  
  
    # Disable caching for real-time updates  
    add_header Cache-Control no-cache;  
    add_header Connection keep-alive;  
    add_header Content-Type text/event-stream;  
  
    # Optional: Set the maximum timeout for SSE to stay connected  
    proxy_read_timeout 3600s; # Keep the connection open for an hour  
  
    access_log /var/log/nginx/sse_access.log;  
}
```

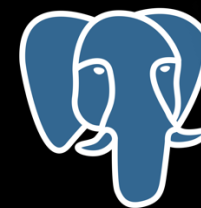


Redis



Pub / Sub Events

- Event Synchronisation
- Grösste Baustelle
- Race Conditions



Postgres



Players / Lobbies



Init-Script

Lobby State

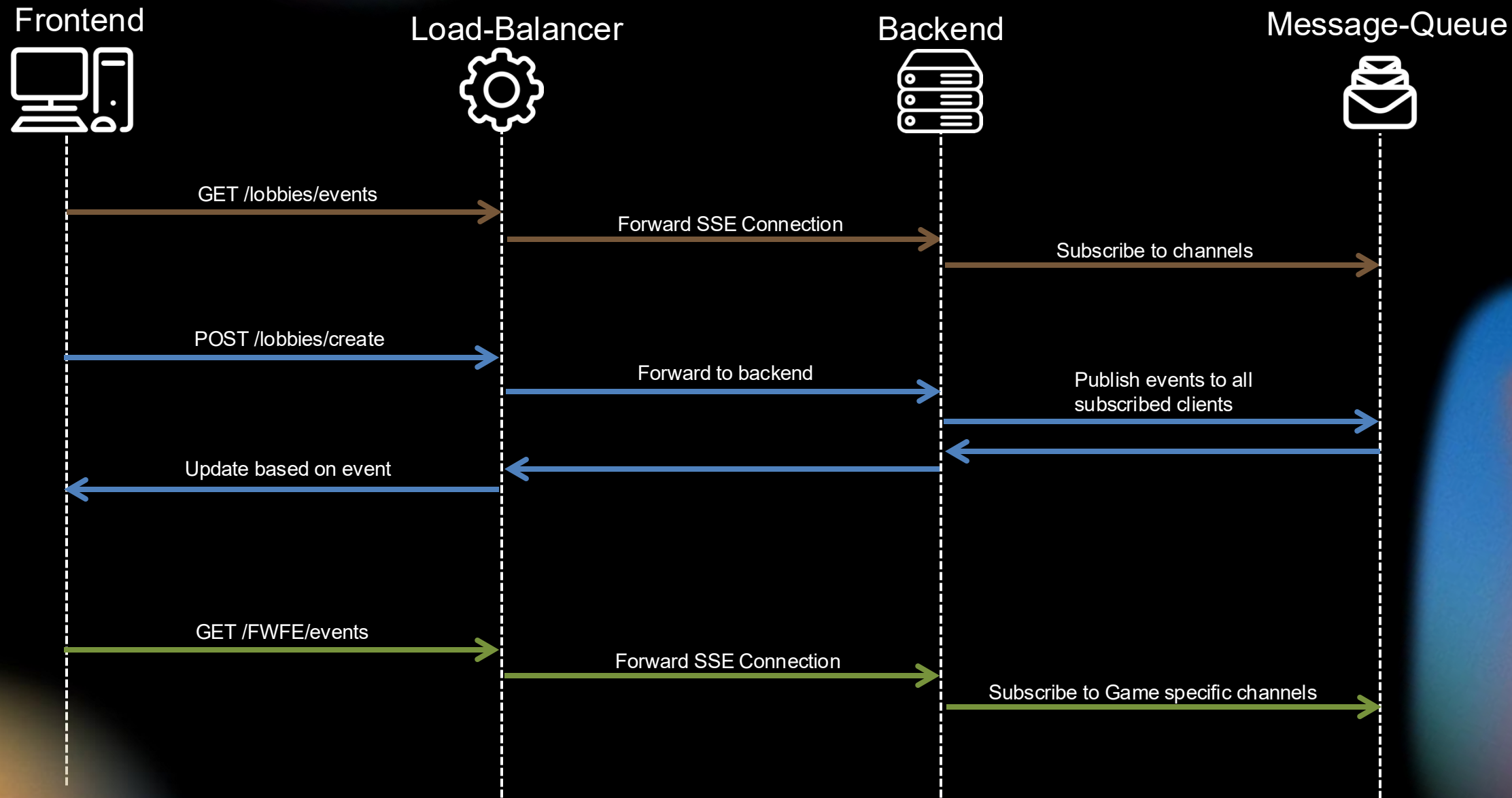


Questions





SSE





Challenges

Docker Setup

Load-Balancer mit SSE / Backends

Multiple Backends

State-Management

So viele Logs



Demo



<http://146.136.163.212:8080>



Demo



Zukunft

Chat-Funktionalität

Community Questions

Mehr Fragentypen

Online hosten



Q&A

