

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бэк-энд разработка

Отчет
Практическая работа №6

Выполнил:

Екушев Владислав

К33402

Проверил:

Добряков Д. И.

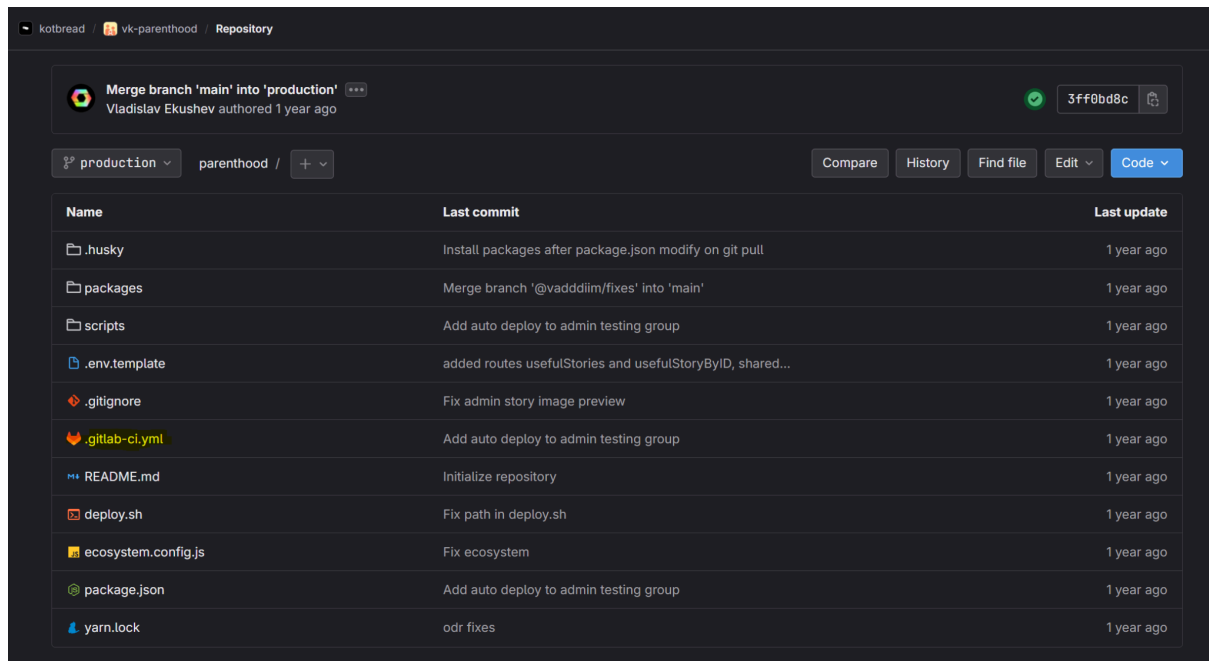
Санкт-Петербург

2024 г.

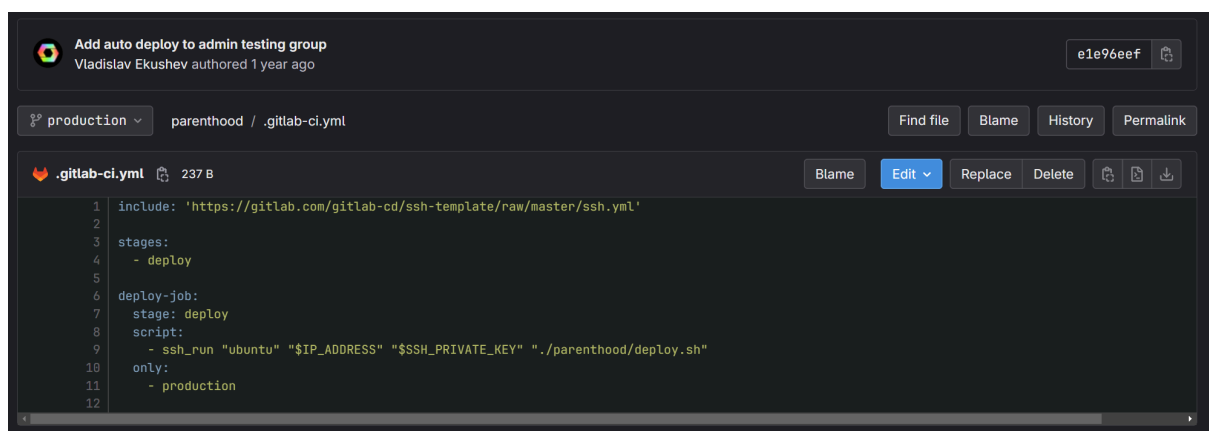
Задача

Необходимо настроить автодеплой (с триггером на обновление кода в вашем репозитории, на определенной ветке) для вашего приложения на удаленный сервер с использованием Github Actions или Gitlab CI (любая другая CI-система также может быть использована).

Ход работы



Имеется рабочий репозиторий для проекта [Родительство](#) со стеком Vite+Express+MongoDB. В нем был создан конфигурационный файл .gitlab-ci.yml с настройкой деплоя:



Используется шаблон для вызова команд по SSH на удаленном сервере. Вызывается команда deploy.sh:

```
deploy.sh 1.19 KiB
Blame Edit Replace Delete
1 #!/bin/bash
2
3 cd parenthood
4
5 echo ""
6 echo "o((o_o))o. Выкладываем на прод..."
7 echo ""
8
9 export PATH=/home/ubuntu/.nvm/versions/node/v19.7.0/bin:/home/ubuntu/bin:/usr/local/bin:/home/ubuntu/.nvm/versions/node/v19.7.0/bin:/home/ubuntu/b
10
11 git fetch origin production
12
13 git checkout production
14
15 git pull
16
17 echo ""
18 echo "📦 Собираем бекенд..."
19
20 yarn workspace @parenthood/backend build
21
22 echo ""
23 echo "📦 Собираем фронтенд..."
24
25 yarn workspace @parenthood/frontend build
26
27 echo "🌟 Проект был успешно собран для выкладки на продакшн!"
28
29 echo "🚀 Запускаем деплой на VK MiniApps Hosting..."
30
31 PROD_URL="$(yes | yarn workspace @parenthood/frontend deploy | awk '$1 == "vk.com:"{print $2}' | sed -n '2 p')"
32
33 yarn update-frontend-webview $PROD_URL
34
35 echo "🔄 Перезагружаем бекенд..."
36
37 pm2 start ecosystem.config.js
38
39 echo "🔥 Проект был успешно обновлён"
```

Видно, что сначала настраивается окружение, подтягивается новый код из репозитория и происходит сборка приложения. Далее скрипт делает деплой на хостинг VK мини-приложений, используя команду `vk-miniapps-deploy`, и вытаскивает из результата выполнения команды строку с актуальным адресом деплоя и обновляет его в настройках приложения командой `“yarn update-frontend-webview”`.

✔ Passed 🕒 00:01:27 📅 1 year ago	#33: deploy-job 🏷️ production 🔑 3ff8bd8c	#151 created by 🧑 Stage: deploy
✔ Passed 🕒 00:01:36 📅 1 year ago	#32: deploy-job 🏷️ production 🔑 99845c48	#121 created by 🧑 Stage: deploy
✔ Passed 🕒 00:03:30 📅 1 year ago	#31: deploy-job 🏷️ production 🔑 68b8fc0f	#105 created by 🧑 Stage: deploy

Для проверки МРов, настроен специальный stage сервер, на котором запускается тестовая версия приложения из какой-либо ветки. Так выглядит `.gitlab-ci.yml` для stage сервера:

```
79dc303c23fb54bd46...  parenthood / .gitlab / .gitlab-ci.yml  Find file  Blame  History  Permalink

.gitlab-ci.yml 759 B  Blame  Edit  Replace  Delete  Copy  Download

1 stage-server:
2   stage: deploy
3   image: alpine
4   before_script:
5     - apk add openssh-client
6     - eval $(ssh-agent -s)
7     - echo "$STAGE_SSH_KEY" | tr -d '\n' | ssh-add -
8     - mkdir -p ~/.ssh
9     - chmod 700 ~/.ssh
10
11   script:
12     - echo "Start pipeline"
13     - |
14       ssh -o StrictHostKeyChecking=no "$STAGE_USER@$STAGE_HOST" -p "$STAGE_PORT" bash <<EOF
15         source /root/.bashrc
16         source ~/.nvm/nvm.sh;
17         echo "cd $STAGE_PATH"
18         cd $STAGE_PATH
19         git reset --hard
20         git fetch
21         git checkout $CI_COMMIT_REF_NAME
22         git cherry-pick $CI_COMMIT_SHA
23         docker compose -f docker-compose.stage.yml up -d --build parenthood-frontend parenthood-backend
24       EOF
25
26   allow_failure: false
27   only:
28     - merge_request
29
```

Успешное выполнение задачи раннером:

kolbredd vk-parenthood jobs #2689

Search job log

```
186 #26 40.67 dist/assets/eruda-41c22b70.js 385.77 kB | gzip: 149.38 kB
187 #26 40.67 dist/assets/index-edd26459.js 1,888.78 kB | gzip: 375.69 kB
188 #26 40.67 ✓ built in 37.81s
189 #26 40.67
190 #26 40.67 (!) Some chunks are larger than 500 kB after minification. Consider:
191 #26 40.67 - Using dynamic import() to code-split the application
192 #26 40.67 - Use build.rollupOptions.output.manualChunks to improve chunking: https://rollupjs.org/configuration-options/#output-manualChunks
193 #26 40.67 - Adjust chunk size limit for this warning via build.chunkSizeWarningLimit.
194 #26 40.83 Done in 39.65s.
195 #26 40.84 Done in 40.14s.
196 #26 DONE 41.8s
197 #27 [parenthood-frontend production 2/4] WORKDIR /parenthood
198 #27 CACHED
199 #28 [parenthood-frontend production 3/4] COPY --from=builder /parenthood/packages/frontend/dist/ /usr/share/nginx/html/.
200 #28 DONE 0.5s
201 #29 [parenthood-frontend production 4/4] COPY ./packages/frontend/nginx.proxy.conf /etc/nginx/conf.d/default.conf
202 #29 DONE 0.3s
203 #30 [parenthood-frontend] exporting to image
204 #30 exporting layers
205 #30 exporting layers 0.2s done
206 #30 writing image sha256:4e320c765f8a2abaf1b6089e07d731345b47bf40d2897f9152f9d6050992aaaa 0.0s done
207 #30 naming to docker.io/library/parenthood-parenthood-frontend 0.0s done
208 #30 DONE 0.3s
209 Container redis Running
210 Container parenthood-frontend Recreate
211 Container parenthood-backend Running
212 Container parenthood-frontend Recreated
213 Container redis Waiting
214 Container parenthood-frontend Starting
215 Container parenthood-frontend Started
216 Container redis Healthy
217 Job succeeded
```

Duration: 4 minutes 18 seconds

Finished: 19 hours ago

Queued: 0 seconds

Timeout: 20m (from runner) ?

Runner: #1 (QJ66FnxmA) CI/CD runner #1

Commit 79dc303c in 185 additional color fixes

Pipeline #1789 Passed for 185 with @Linusce/calcs

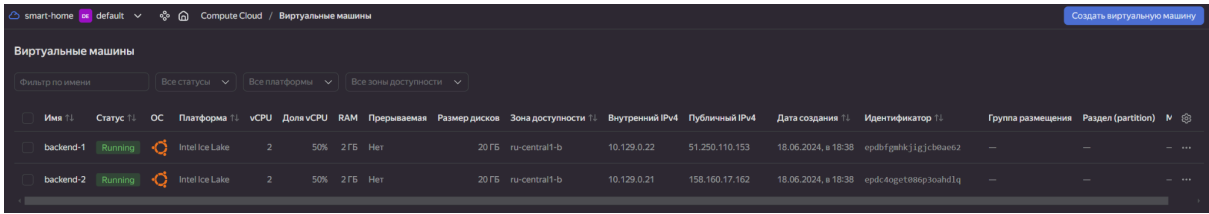
deploy

Related jobs

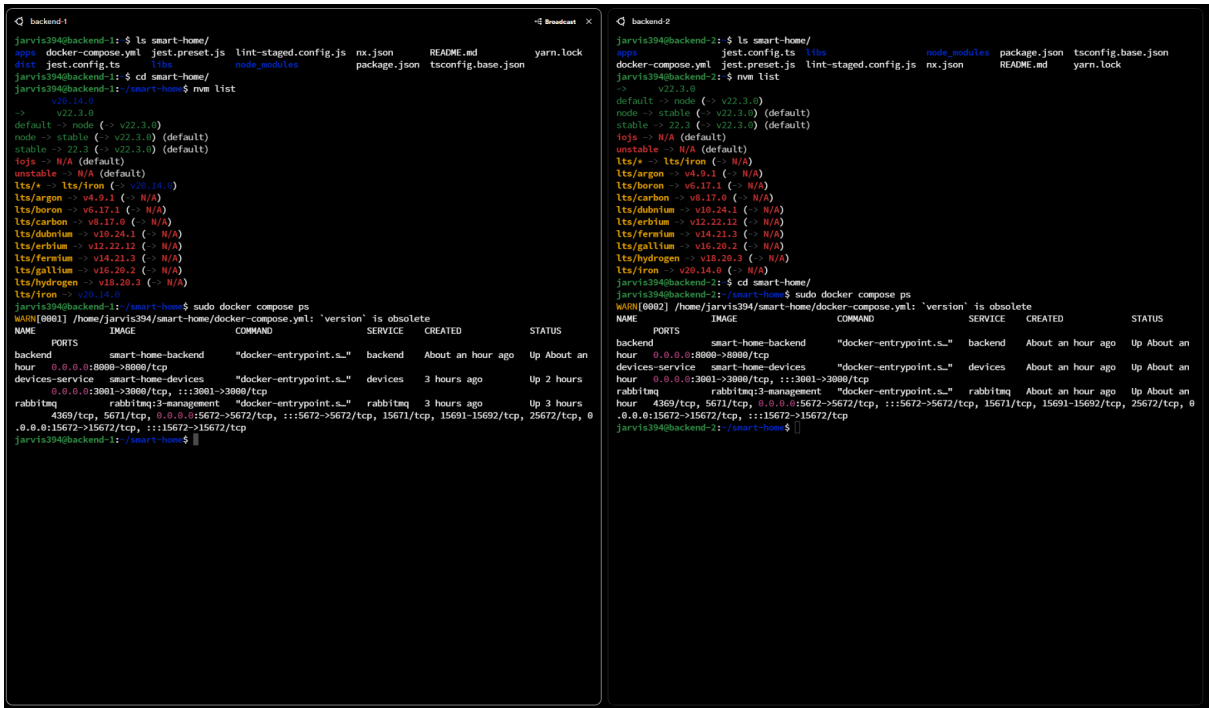
→ stage-server

Load Balancer

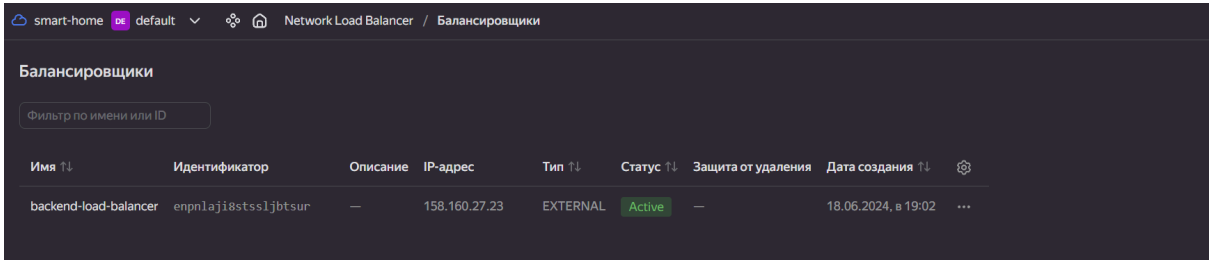
Создадим группу виртуальных машин на Yandex Cloud



Настроим виртуальные машины под использование докера и nginx:



Создадим балансировщик нагрузки:



Зададим целевую группу для балансировщика:

Network Load Balancer / Балансировщики / backend-load-balancer

Остановить Редактировать + Подключить целевую группу

Обзор

Идентификатор

enpn1aji8stss1jbtstur

Статус

Active

Имя

backend-load-balancer

Регион

ru-central1

Тип

EXTERNAL

Дата создания

18.06.2024, в 19:02

Обработчики

Имя	Адрес обработчика ↑↓	Порт	Целевой порт	Протокол ↑↓	
listener-acd9c-db9	158.160.27.23	80	80	TCP	...
listener-a21ef-a1e	158.160.27.23	443	443	TCP	...

Целевые группы

backend

enp197c2z0ep47h42s7p

...

Ресурсы 2

Проверка состояния

ВМ или IP-адрес

--

ВМ	Адрес	Статус ↑↓
backend-2	10.129.0.21	Healthy
backend-1	10.129.0.22	Healthy

Регистрируем домен для балансировщика:

smart-home-api.kotbread.ru

158.160.27.23

...

Создадим сертификат для HTTPS трафика к балансировщику:

```
jarvis394@backend-1:/etc/nginx$ sudo certbot --nginx
Saving debug log to /var/log/letsencrypt/letsencrypt.log

Which names would you like to activate HTTPS for?
We recommend selecting either all domains, or all domains in a VirtualHost/server block.
-----
1: smart-home-api.kotbread.ru
-----
Select the appropriate numbers separated by commas and/or spaces, or leave input
blank to select all options shown (Enter 'c' to cancel): 1
Requesting a certificate for smart-home-api.kotbread.ru

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/smart-home-api.kotbread.ru/fullchain.pem
Key is saved at: /etc/letsencrypt/live/smart-home-api.kotbread.ru/privkey.pem
This certificate expires on 2024-09-16.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for smart-home-api.kotbread.ru to /etc/nginx/sites-enabled/default
Congratulations! You have successfully enabled HTTPS on https://smart-home-api.kotbread.ru

-----
If you like Certbot, please consider supporting our work by:
* Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
* Donating to EFF: https://eff.org/donate-le
-----
```

Также нужно вставить тот же сертификат на второй сервер.

Поднимаем бесплатный кластер MongoDB:

All Clusters

Availability All

Type All

Version All

Configuration All

Search for names and tags...

Q

Showing 1 of 6 clusters.

Show Inactive Clusters

jarvis394/smart-home

Atlas

Name	Version	Data Size	Nodes	Backup	SSL	Auth	Alerts
<div><div></div><div>main</div></div>	7.0.11	N/A	3	OFF	ON	ON	

Разрешаем доступ с IP адресов серверов:

JARVIS394 > SMART-HOME

Network Access

IP Access List

Peering

Private Endpoint

+ ADD IP ADDRESS

You will only be able to connect to your cluster from the following list of IP Addresses:

IP Address	Comment	Status	Actions
158.160.93.207/32	backend-2	<div></div> Active	<div>EDIT</div> <div>DELETE</div>
158.160.68.29/32	backend-1	<div></div> Active	<div>EDIT</div> <div>DELETE</div>

Делаем деплой фронтенд части приложения:

Project Name

Used to identify your Project on the Dashboard, Vercel CLI, and in the URL of your Deployments.

vercel.com/jarvis394s-projects/

smart-home

Learn more about Project Name

Save

Build & Development Settings

When using a framework for a new project, it will be automatically detected. As a result, several project settings are automatically configured to achieve the best result. You can override them below.

Framework Preset

Vite

Build Command

npx nx run frontend:build:production

Override

Output Directory

dist/apps/frontend

Override

Install Command

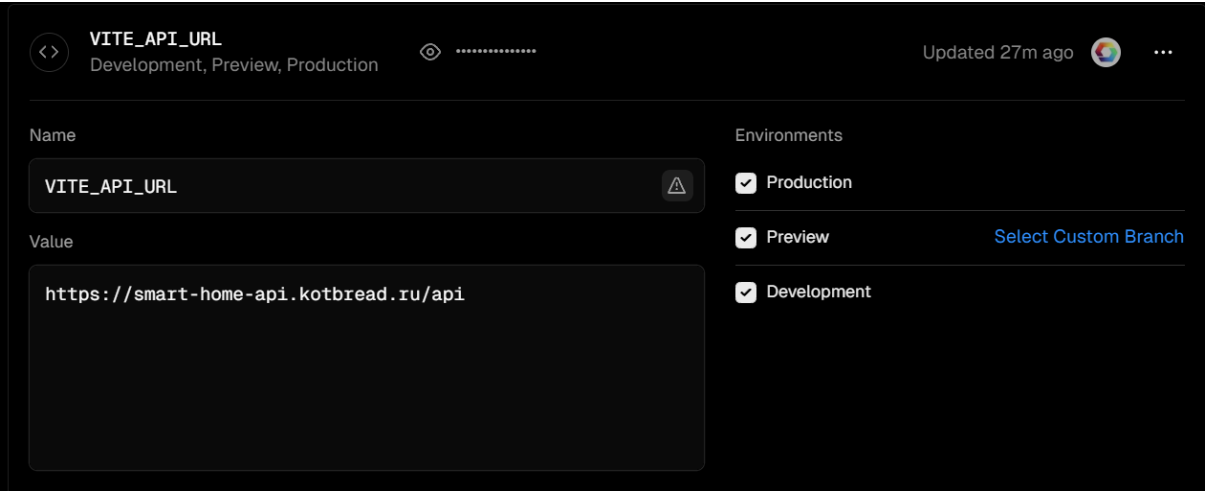
yarn install

Override

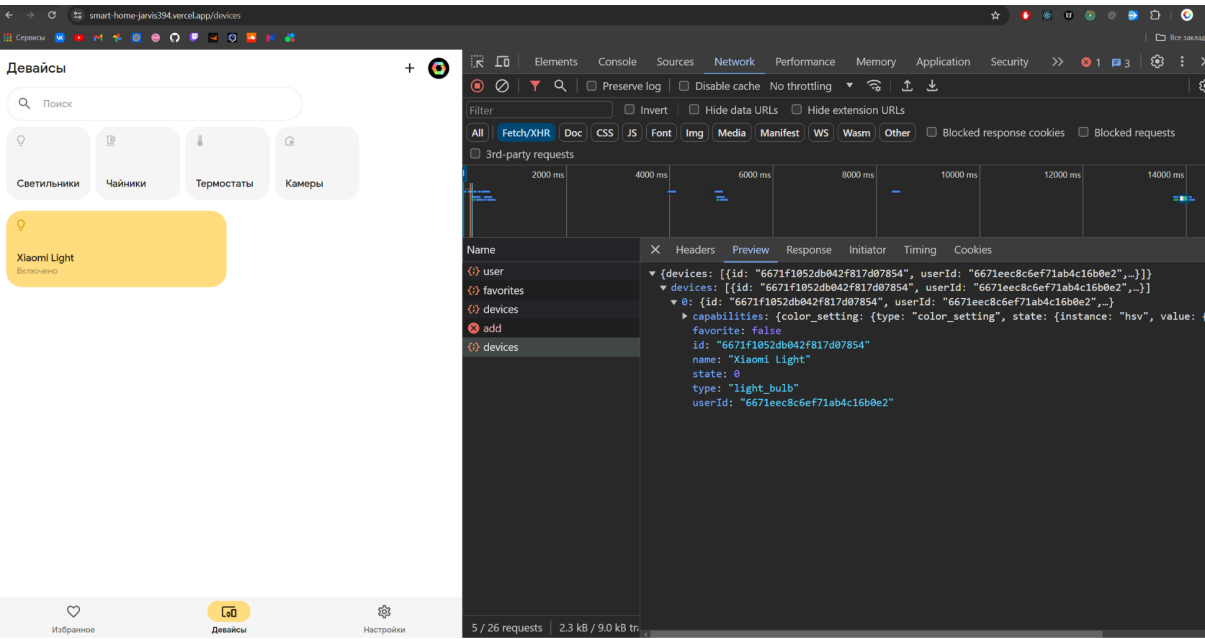
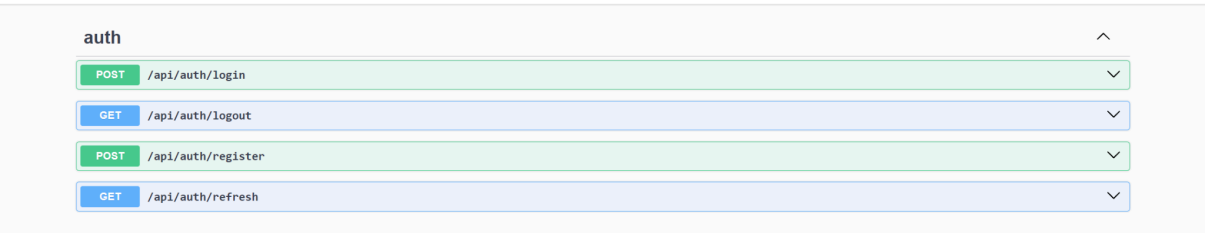
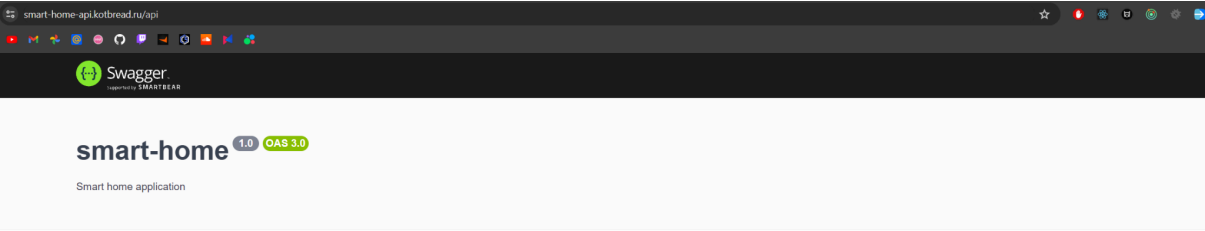
Development Command

vite

Override



Результат:



Вывод

В ходе работы были получены навыки работы с Yandex Cloud и удаленного развертывания.

Фронтенд – <https://smart-home-jarvis394.vercel.app/>

Бекенд – <https://smart-home-api.kotbread.ru/api>