

отчёт о выполнении курсовой работы

установка и настройка сервисов Gitea / Jenkins

Работу выполнил
студент ИВТ-31 БО
Головач Артем

21.06.2021

1. структура docker-compose

```
version: '3.2'
services:
  jenkins:
    container_name: jenkins
    build:
      context: jenkins
      dockerfile: Dockerfile
      args:
        JENKINS_VERSION: 2.235
    ports:
      - 8080:8080
      - 50000:50000
    volumes:
      - /tmp:/var/jenkins_home/secrets/
  gitea:
    container_name: gitea
    build:
      context: gitea
    ports:
      - 222:22
      - 3000:3000
```

- На каждый компонент приходится свой компонент services.
- context указывает на отдельную директорию с клонированным официальным репозиторием проекта
- dockerfile указывает на необходимый Dockerfile для сборки
- у Jenkins указана стабильная версия
- директива volumes (Jenkins) нужна для получения секретного ключа, необходимого для успешного входа при первом запуске
- приложения работают на портах по умолчанию:
jenkins - 8080
gitea - 3000 (222 - при использовании SSH)

2. описание Dockerfile

a) Jenkins

в качестве родительского образа указан openjdk:8-jdk-alpine, так как jenkins написан преимущественно на Java

```
FROM openjdk:8-jdk-alpine
```

установка в зависимостей

```
RUN apk add --no-cache \  
    bash \  
    coreutils \  
    curl \  
    git \  
    git-lfs \  
    openssh-client \  
    tini \  
    ttf-dejavu \  
    tzdata \  
    unzip \  
    python3
```

задаём переменные для передачи Docker во время сборки образа
устанавливаем постоянные переменные среды

```
ARG user=jenkins  
ARG group=jenkins  
ARG uid=1000  
ARG gid=1000  
ARG http_port=8080  
ARG agent_port=50000  
ARG JENKINS_HOME=/var/jenkins_home  
ARG REF=/usr/share/jenkins/ref  
  
ENV JENKINS_HOME $JENKINS_HOME  
ENV JENKINS_SLAVE_AGENT_PORT ${agent_port}  
ENV REF $REF
```

Создание корневого каталога , добавление группы и пользователя, изменение владельца корневого каталога

```
RUN mkdir -p $JENKINS_HOME \  
  && chown ${uid}:${gid} $JENKINS_HOME \  
  && addgroup -g ${gid} ${group} \  
  && adduser -h "$JENKINS_HOME" -u ${uid} -G ${group} -s /bin/bash -D ${user}
```

создаём точку монтирования для работы с постоянным хранилищем

```
VOLUME $JENKINS_HOME
```

Скачивание исполняемого файла jenkins.war

```
RUN curl -fsSL ${JENKINS_URL} -o /usr/share/jenkins/jenkins.war \  
  && echo "${JENKINS_SHA} /usr/share/jenkins/jenkins.war"
```

открытие портов

```
# for main web interface:  
EXPOSE ${http_port}  
  
# will be used by attached slave agents:  
EXPOSE ${agent_port}
```

добавление скриптов для запуска приложения

```
COPY jenkins-support /usr/local/bin/jenkins-support  
COPY jenkins.sh /usr/local/bin/jenkins.sh  
COPY tini-shim.sh /bin/tini  
ENTRYPOINT ["/sbin/tini", "--", "/usr/local/bin/jenkins.sh"]
```

добавление скриптов для плагинов

```
COPY plugins.sh /usr/local/bin/plugins.sh  
COPY install-plugins.sh /usr/local/bin/install-plugins.sh
```

б) Gitea

в качестве родительского образа указан golang:1.16-alpine3.13, так как gitea написан преимущественно на GO

```
FROM golang:1.16-alpine3.13 AS build-env
```

задаём переменные для передачи Docker во время сборки образа
устанавливаем постоянные переменные среды

```
ARG GOPROXY
```

```
ENV GOPROXY ${GOPROXY:-direct}
```

```
ARG GITEA_VERSION
```

```
ARG TAGS="sqlite sqlite_unlock_notify"
```

```
ENV TAGS "bindata timetzdata $TAGS"
```

```
ARG CGO_EXTRA_CFLAGS
```

установка в зависимостей

```
#install deps
```

```
RUN apk --no-cache add build-base git nodejs npm
```

установка рабочей директории для репозитория

```
#workdir
```

```
COPY . ${GOPATH}/src/code.gitea.io/gitea
```

```
WORKDIR ${GOPATH}/src/code.gitea.io/gitea
```

проверка наличия версии, если такая стоит

```
RUN if [ -n "${GITEA_VERSION}" ]; then git checkout "${GITEA_VERSION}"; fi \
    && make clean-all build
```

начало сборки env-to-ini

```
RUN go build contrib/environment-to-ini/environment-to-ini.go
```

описываем метаданные

```
FROM alpine:3.13
```

```
LABEL maintainer="maintainers@gitea.io"
```

открытие портов

```
EXPOSE 22 3000
```

задаём переменные для передачи Docker во время сборки образа
устанавливаем постоянные переменные среды

```
RUN apk --no-cache add \  
    bash \  
    ca-certificates \  
    curl \  
    gettext \  
    git \  
    linux-pam \  
    openssh \  
    s6 \  
    sqlite \  
    su-exec \  
    gnupg  
  
RUN addgroup \  
    -S -g 1000 \  
    git && \  
adduser \  
    -S -H -D \  
    -h /data/git \  
    -s /bin/bash \  
    -u 1000 \  
    -G git \  
    git && \  
    echo "git:*" | chpasswd -e  
  
ENV USER git  
ENV GITEA_CUSTOM /data/gitea
```

создание входной точки

```
CMD ["/bin/s6-svscan", "/etc/s6"]
```

```
COPY --from=build-env /go/src/code.gitea.io/gitea/gitea /app/gitea/gitea
```

```
COPY --from=build-env /go/src/code.gitea.io/gitea/environment-to-ini /usr/local/bin/environment-to-ini
```

```
RUN ln -s /app/gitea/gitea /usr/local/bin/gitea
```

```
QA user@debian: ~/final_exam/unix_exam
```

```
root@debian:/final_exam/unix_exam# cat /var/log/jenkins/jenkins.log
jenkins      2021-06-28 17:41:33.840+0000 [id=26] INFO     jenkins.InitReactorRunner$1onAttained: Prepared all plugins
jenkins      2021-06-28 17:41:34.873+0000 [id=26] INFO     jenkins.InitReactorRunner$1onAttained: Started all plugins
jenkins      2021-06-28 17:41:34.935+0000 [id=25] INFO     jenkins.InitReactorRunner$1onAttained: Augmented all extensions
jenkins      2021-06-28 17:41:35.275+0000 [id=26] INFO     jenkins.InitReactorRunner$1onAttained: System config loaded
jenkins      2021-06-28 17:41:38.722+0000 [id=26] INFO     jenkins.InitReactorRunner$1onAttained: System config adapted
jenkins      2021-06-28 17:41:38.725+0000 [id=26] INFO     jenkins.InitReactorRunner$1onAttained: Loaded all jobs
jenkins      2021-06-28 17:41:38.735+0000 [id=25] INFO     jenkins.InitReactorRunner$1onAttained: Configuration for all jobs updated
jenkins      2021-06-28 17:41:38.802+0000 [id=39] INFO     hudson.model.AsyncPeriodicWork lambda$doRun$: Starting download metadata
jenkins      2021-06-28 17:41:38.803+0000 [id=39] INFO     hudson.util.ActionStart: Attempt #1 to do the action check updates server
jenkins      2021-06-28 17:41:43.691+0000 [id=25] INFO     o.s.c.s.AbstractApplicationContext$prepareRefresh: Refreshing org.springframework.web.context.support.StaticWebApplicationContext@cf4746a: display name [Root WebApplicationContext]; startup date [Sun Jun 28 17:41:43 GMT 2021]; root of context hierarchy
jenkins      2021-06-28 17:41:43.693+0000 [id=25] INFO     o.s.c.s.AbstractApplicationContext$obtainFreshBeanFactory: Bean factory for application context [org.springframework.web.context.support.StaticWebApplicationContext@cf4746a]: org.springframework.beans.factory.support.DefaultListableBeanFactory@8d40a75
jenkins      2021-06-28 17:41:43.825+0000 [id=25] INFO     o.s.b.f.s.DefaultListableBeanFactory#preInstantiateSingletons: Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@8d40a75: defining beans [authenticationManager]; root of factory hierarchy
jenkins      2021-06-28 17:41:45.264+0000 [id=25] INFO     o.s.c.s.AbstractApplicationContext$prepareRefresh: Refreshing org.springframework.web.context.support.StaticWebApplicationContext@5f0434d3: display name [Root WebApplicationContext]; startup date [Sun Jun 28 17:41:45 GMT 2021]; root of context hierarchy
jenkins      2021-06-28 17:41:45.275+0000 [id=25] INFO     o.s.c.s.AbstractApplicationContext$obtainFreshBeanFactory: Bean factory for application context [org.springframework.web.context.support.StaticWebApplicationContext@5f0434d3]: org.springframework.beans.factory.support.DefaultListableBeanFactory@6b385529
jenkins      2021-06-28 17:41:45.282+0000 [id=25] INFO     o.s.b.f.s.DefaultListableBeanFactory#preInstantiateSingletons: Pre-instantiating singletons in org.springframework.beans.factory.support.DefaultListableBeanFactory@6b385529: defining beans [filter,legacy]; root of factory hierarchy
jenkins      2021-06-28 17:41:47.561+0000 [id=25] INFO     jenkins.install.SetupWizard#init:
jenkins
jenkins
*****
jenkins
*****
jenkins
*****
jenkins
Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:
jenkins
*****
jenkins
*****
jenkins
This may also be found at: /var/jenkins_home/secrets/initialAdminPassword
jenkins
*****
jenkins
*****
jenkins
*****
jenkins
*****
jenkins
*****
jenkins      2021-06-28 17:42:58.293+0000 [id=39] INFO     h.m.DownloadService$downloadable$load: Obtained the updated data file for hudson.tasks.Maven.MavenInstaller
jenkins      2021-06-28 17:42:58.300+0000 [id=39] INFO     hudson.util.ActionStart: Performed the action check updates server successfully at the attempt #1
jenkins      2021-06-28 17:42:58.325+0000 [id=39] INFO     hudson.model.AsyncPeriodicWork lambda$doRun$: Finished Download metadata. 79,311 ms
jenkins      2021-06-28 17:43:01.040+0000 [id=26] INFO     jenkins.InitReactorRunner$1onAttained: Completed initialization
jenkins      2021-06-28 17:43:01.156+0000 [id=19] INFO     jenkins.WebAppMain$3run: Jenkins is fully up and running
gitlab       2021/06/28 17:43:21 ...rs/common/logger.go:211:10 | I | Started GET / for 192.168.1.66:49921
gitlab       2021/06/28 17:43:21 ...nt/context.go:234:10:10 | O | Template: install
gitlab       2021/06/28 17:43:21 ...rs/common/logger.go:301:10 | I | Completed GET / 200 OK in 3.500241ms
gitlab       2021/06/28 17:43:21 ...rs/common/logger.go:211:10 | I | Started GET /assets/css/index.css?v=d4108cd8f00b204e98009998cB847ze for 192.168.1.66:49921
gitlab       2021/06/28 17:43:21 ...rs/common/logger.go:301:10 | I | Completed GET /assets/css/index.css?v=d4108cd8f00b204e98009998cB847ze 200 OK in 3.277487ms
gitlab       2021/06/28 17:43:21 ...rs/common/logger.go:211:10 | I | Started GET /assets/img/loading.png for 192.168.1.66:49922
gitlab       2021/06/28 17:43:21 ...rs/common/logger.go:301:10 | I | Completed GET /assets/img/loading.png for 192.168.1.66:49922
```

Getting Started

Create First Admin User

Имя пользователя:

Пароль:

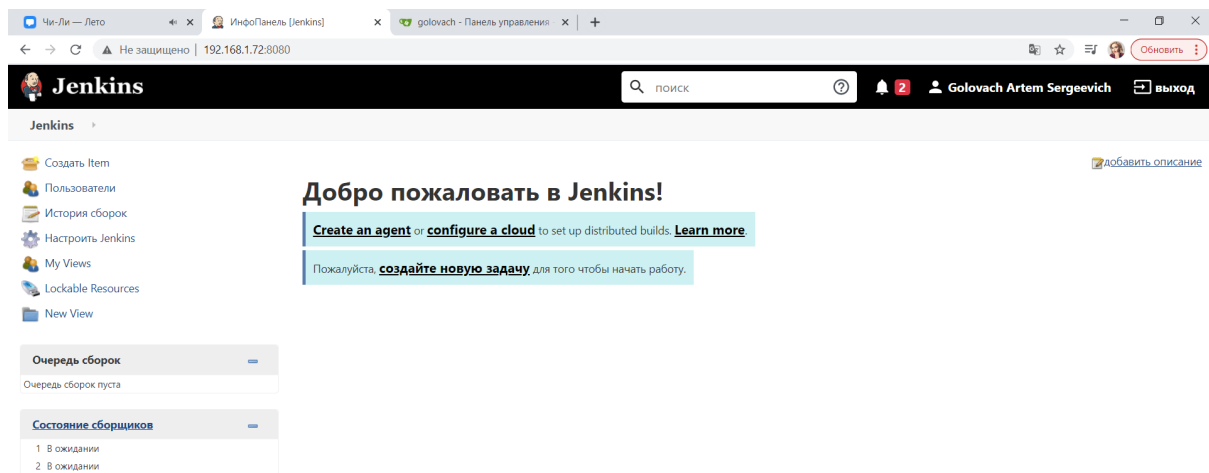
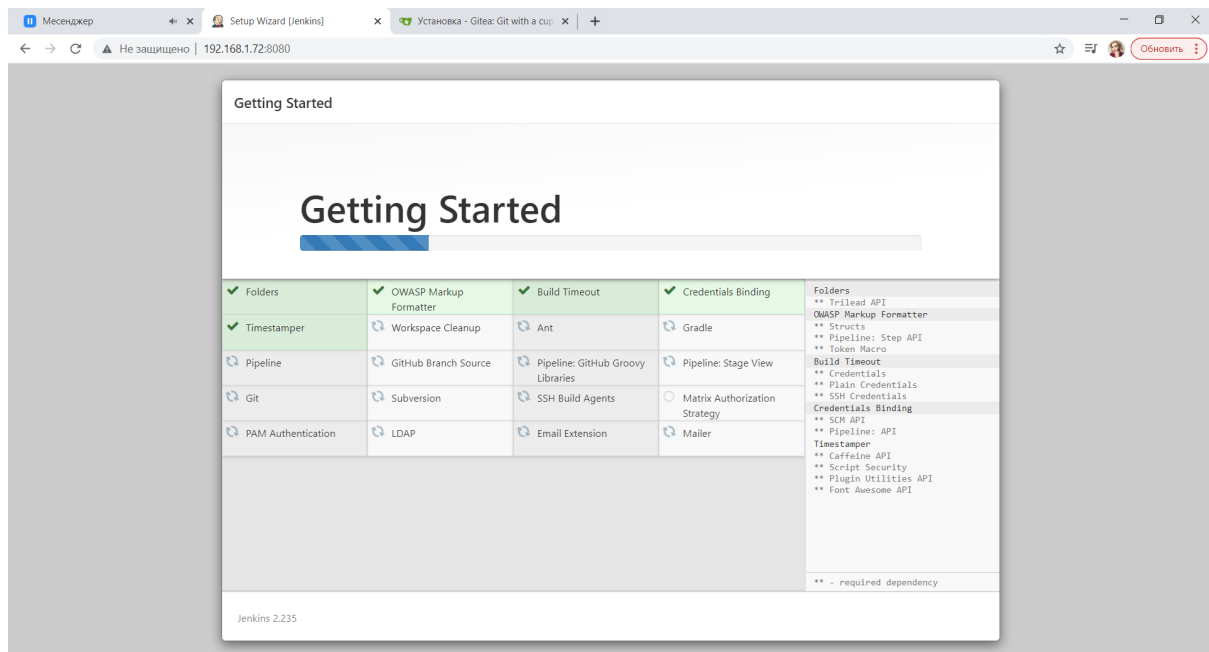
Повторите пароль:

Ф.И.О.:

Адрес электронной почты:

Jenkins 2.235

[Skip and continue as admin](#) [Save and Continue](#)



После первого входа в Jenkins и ввода секретного ключа, пользователю предоставлена возможность установить плагины по умолчанию или выбрать необходимые. Я воспользовался настройкой по умолчанию, потому что она включает плагин git, необходимый для успешной интеграции с Gitea

Мессенджер

Setup Wizard [Jenkins]

Установка - Gitea: Git with a cup of

+

← → ↻ Не защищено | 192.168.1.72:3000

Обновить

Запуск от имени пользователя *

git

Введите имя пользователя операционной системы, под которым работает Gitea. Обратите внимание, что этот пользователь должен иметь доступ к корневому пути репозитория.

Домен SSH сервера *

192.168.1.72

Влияет на URL-адреса для клонирования по SSH.

Порт SSH сервера

222

Номер порта, который использует SSH сервер. Оставьте пустым, чтобы отключить SSH.

Gitea HTTP порт *

3000

Номер порта, который будет прослушиваться Gitea веб-сервером.

Базовый URL-адрес Gitea *

http://192.168.1.72:3000/

Этот параметр влияет на URL для клонирования по HTTP/HTTPS и на некоторые уведомления по электронной почте.

Путь к журналу *

/data/gitea/loq

Файлы журнала будут записываться в этот каталог.

Расширенные настройки

▶ Настройки электронной почты

▶ Сервер и настройки внешних служб

▶ Настройки учётной записи администратора

при запуске Gitea будут предложены настройки по умолчанию, в которых было изменено:

домен SSH сервера изменён с localhost на IP адрес машины

порт SSH сервера изменён на 222

в базовом URL адреса localhost изменён на IP адрес машины

Чи-Ли — Лето

ИнфоПанель [Jenkins]

golovach - Панель управления

+

← → ↻ Не защищено | 192.168.1.72:3000

Панель управления

Задачи

Pull Request'ы

Этапы

Обзор

Обновить

golovach

Учётная запись была успешно создана.

Репозиторий

Организация

Репозитории 0

+

Поиск репозитория...

Все 1

Собственные

Форки

Зеркала

Совместные

Работает на Gitea

Версия: 1.16.5

Страница: 24ms

Шаблон: 16ms

русский

Лицензии


API

Веб-сайт

Go1.16.5

Новый репозиторий

Репозиторий содержит все файлы проекта, включая историю ревизий. Уже есть где-то еще? Мигрировать репозиторий.

Владелец *  golovach

Некоторые организации могут не отображаться в раскрывающемся списке из-за максимального ограничения количества репозиториях.

Название репозитория * test

Лучшие названия репозиториях состоят из коротких, легко запоминаемых и уникальных ключевых слов.

Видимость ☐ Сделать репозиторий приватным

Только владелец или члены организации, при наличии прав, смогут увидеть это.

Описание

Добавьте краткое описание (необязательно)

Шаблон

Выберите шаблон.

Метки задач

Выберите набор ярлыков задачи.

.gitignore

Выберите шаблон .gitignore.

для создания нового репозитория необходимо лишь задать его имя

golovach / test

Перестать следить 1 ☆ в избранное 0 Форкнуть 0

<> Код ① Задачи Pull Request'ы Проекты Релизы Вики Активность Настройки

Нет описания

Редактировать тематические метки

1 коммит 1 ветка 0 Теги 21 KiB

ветка: master Новый Pull Request

Новый файл Загрузить файл HTTP SSH http://192.168.1.72:3000/golovach

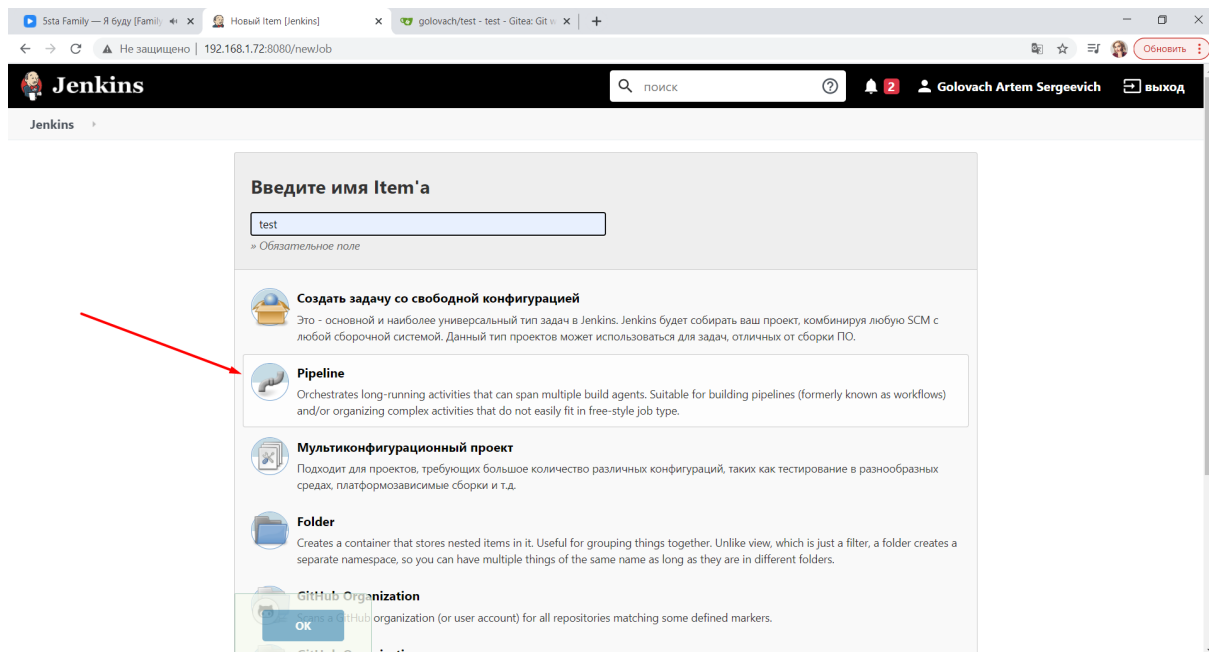
AGolovach	7f272f2703	2nd	33 секунд назад
Jenkinsfile		2nd	33 секунд назад
README.md		2nd	33 секунд назад
ruapp.py		2nd	33 секунд назад
rubyapp.rb		2nd	33 секунд назад

README.md

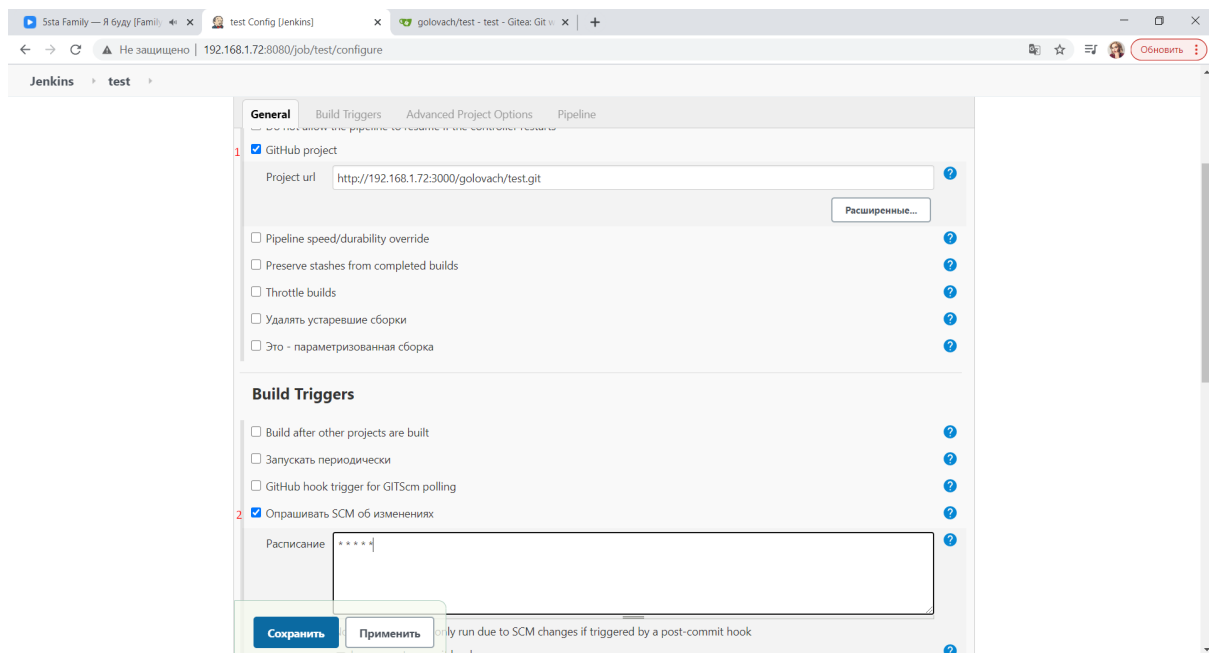
Работает на Gitea Версия: Страница: 76ms Шаблон: 20ms

русский Лицензии API Веб-сайт Go1.16.5

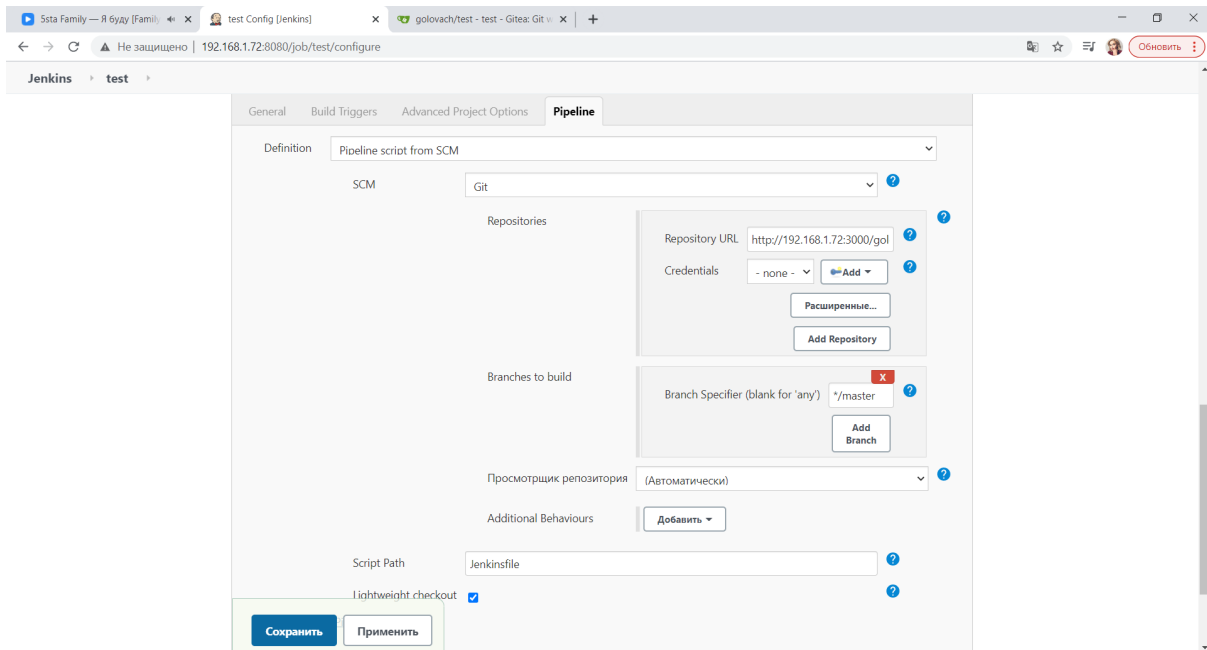
в репозиторий была выгружена директория с заранее заготовленным Jenkinsfile запускающим ruapp.py



для создания нового item'a сборки в Jenkins переходим в раздел Pipeline



ставим галочку на “GitHub project” и указываем наш путь к репозиторию
ставим галочку на “Опрашивать SCM об изменениях” и вводим * * * * *, что
равносильно проверке изменений каждую минуту



в разделе настройки Pipeline указываем “Pipeline from SCM” и путь до нашего репозитория, а также путь скрипта, если он имеет название отличное от Jenkinsfile сохраняем изменения

Pipeline test

[Recent Changes](#)

Stage View

Declarative: Checkout SCM	Build	Test	Deploy
1s	1s	283ms	285ms
1s	1s	283ms	285ms

Average stage times: (Average full run time: ~36s)

Постоянные ссылки

История сборок: [Тренд](#)

find

20.06.2021 18:03

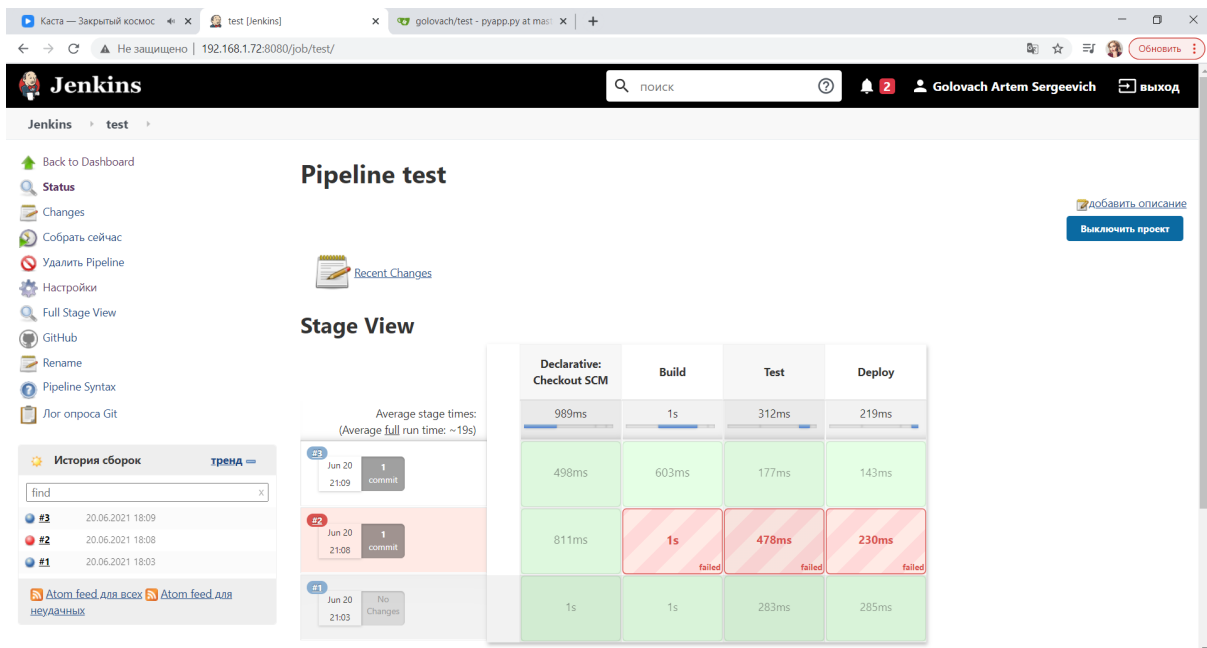
Atom feed для всех Atom feed для неулучшенных

Страница сгенерирована: 20.06.2021 18:02:41 GMT REST API Jenkins 2.235

```
Jenkins > test > #1

[Pipeline] withEnv
[Pipeline] {
[Pipeline] timestamps
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
18:03:41 Building..
[Pipeline] sh
18:03:42 + python3 pyapp.py
18:03:42 Hello, World!
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] echo
18:03:42 Testing..
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] echo
18:03:42 Deploying....
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // timestamps
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Страница сгенерирована: 20.06.2021 18:04:50 GMT REST API Jenkins 2.235



пример успешной сборки проекта, а также неуспешной в случае изменения нашего .py файла

