Ch-134.DevOps/Cisco

Demo 3

IT Academy

softserve

Team 2

Ivan Kuvila
Pavlo Tarnovetskyi
Ivan Danyliuk
Vladyslav Boreiko

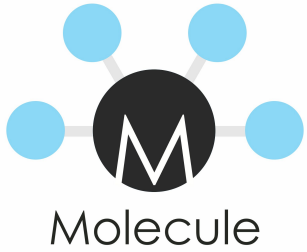
Ch-134.DevOps/Cisco

softserve

https://github.com/wladOSnull/SSITA_DevOps

Vladyslav Boreiko

Tasks for demonstration:



- Go
  - SSITA: Golang Core
  - GitHub: GolangUA worksł
- Supervisor
- Molecule
- Elastic Stack
- GitLab CI/CD
  - Dockerize Geo Citizen + Supervisor
  - Sample deploy

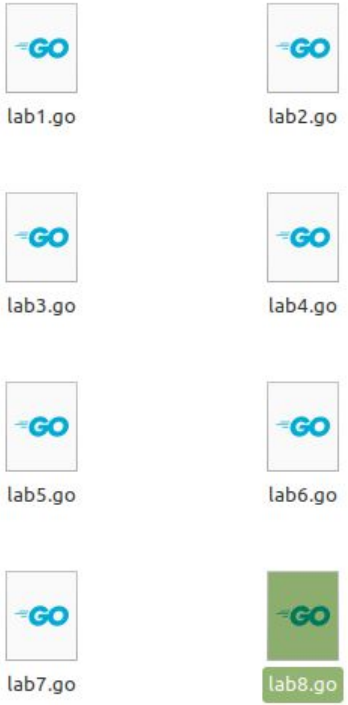
softserve

Go: SSITA

Golang Core:

- 8 tasks:
  - basic io
  - basic types
  - statements
  - loops
  - functions
- quiz

```go
package main

import (
    "fmt"
    "os"
    "strconv"
)

func getMinMax(a []string) (int, int) {

    max, _ := strconv.Atoi(a[0])
    min := max

    for _, v := range a {

        v, _ := strconv.Atoi(v)
        if v > max {
            max = v
        }

        if v < min {
            min = v
        }
    }

    return min, max
}

func main() {

    args := os.Args[1:]

    a1, a2 := getMinMax(args)

    fmt.Printf("min: %d\nmax: %d\n", a1, a2)
}
```



softserve

Vladyslav Boreiko

https://github.com/GolangUA/workshops

Go: GitHub

GitHub: GolangUA workshops:

- Beginner
  - Print chessboard
  - Scan positive&even number from string
  - Bank card validator (regexp)
  - Fibonacci closure
  - Palindrome
  - Lucky ticket
- Mid
  - Simple JSON parser

```
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon$ go run task0.go
width <space> heigth <space> symbol: 10 4 ^

parameters: 10 4 ^

^ ^ ^ ^ ^ ^ ^ ^ ^ ^
 ^ ^ ^ ^ ^ ^ ^ ^ ^ ^
^ ^ ^ ^ ^ ^ ^ ^ ^ ^
 ^ ^ ^ ^ ^ ^ ^ ^ ^ ^
```

```
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon$ go run task1.go
string of numbers over coma: 30,-1,-6,90,-6,22,52,123,2,35,6

splitted result: [30 -1 -6 90 -6 22 52 123 2 35 6]

positive even numbers: 30, 90, 22, 52, 2, 6,
total of positive even numbers: 6
```

```
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon$ go run task2.go
check card number: 4585 9119 2222 6084

card IS valide
```

```
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon$ go run task3.go
length of Fibonacci sequence: 20
[1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765]
```

softserve

Go: GitHub

GitHub: GolangUA workshops

- Beginner
  - Print chessboard
  - Scan positive&even number from string
  - Bank card validator (regexp)
  - Fibonacci closure
  - Palindrome
  - Lucky ticket
- Medium
  - Simple JSON parser

```
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon$ go run task4.go
input a string: lolasd12321zxcvv

total: 4
list of subpalindroms:

a       b       palindrom
0       2       [l o l]
6       10      [1 2 3 2 1]
7       9       [2 3 2]
14      15      [v v]
```

```
Min: 120123
Max: 320320

--Result--
EasyFormula: 11187
HardFormula: 5790
```

```
--Result--
EasyFormula:  11197
HardFormula:  5790
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon$
```

softserve

Go: GitHub

GitHub: GolangUA workshops

- Beginner
  - Print chessboard
  - Scan positive&even number from string
  - Bank card validator (regexp)
  - Fibonacci closure
  - Palindrome
  - Lucky ticket
- Mid
  - Simple JSON parser

```
.
├── file.json
├── jsonParser
├── jsonParser.go
├── personsOutput.json
├── placesOutput.json
└── thingsOutput.json

0 directories, 6 files
```

```
var jsonStr = []byte(`
{
    "things": [
        {
            "name": "Alice",
            "age": 37
        },
        {
            "city": "Ipoh",
            "country": "Malaysia"
        },
        {
            "name": "Bob",
            "age": 36
        },
        {
            "city": "Northampton",
            "country": "England"
        },
        {
            "name": "Albert",
            "age": 3
        },
        {
            "city": "Dnipro",
            "country": "Ukraine"
        },
        {
            "name": "Roman",
            "age": 32
        },
        {
            "city": "New York City",
            "country": "US"
        }
    ]
}`)
```

```
{
    "things": [
        {
            "persons": [
                {
                    "name": "Albert",
                    "age": 3
                },
                {
                    "name": "Roman",
                    "age": 32
                },
                {
                    "name": "Bob",
                    "age": 36
                },
                {
                    "name": "Alice",
                    "age": 37
                }
            ]
        },
        {
            "places": [
                {
                    "city": "Ipoh",
                    "country": "Malaysia"
                },
                {
                    "city": "Dnipro",
                    "country": "Ukraine"
                },
                {
                    "city": "Northampton",
                    "country": "England"
                },
                {
                    "city": "New York City",
                    "country": "US"
                }
            ]
        }
    ]
}
```

```
wlados@DELL-G7-7588:~/Documents/SSITA/Milestone3/go_marathon/parser$ ./jsonParser file.json
INFO: [{Albert 3} {Roman 32} {Bob 36} {Alice 37}]
INFO: [{Ipoh Malaysia} {Dnipro Ukraine} {Northampton England} {New York City US}]
```

softserve

Supervisor: files

Full dockerization of Geo Citizen:

- Alpine - 3.15
- Tomcat - 9
- PostgreSQL - 14.2
- Java - 11
- Supervisord - 4.2.2
- Geo Citizen

```
# main
FROM alpine:latest
MAINTAINER wlados

# for confguring image
COPY --chmod=777 ./build.sh /home/build.sh
RUN ./home/build.sh

# for PostgreSQL connection
COPY --chmod=777 ./postgres_custom.sh /usr/local/bin/postgres_custom.sh

# for configuring supervisor daemon
COPY ./supervisord.conf /etc/supervisor.d/supervisord.ini

# for deploying Geo Citizen
COPY ./citizen.war /opt/tomcat/latest/webapps/citizen.war

# for Tomcat, Supervisord Web interface
EXPOSE 8080 9001

# for starting supervisor daemon as main process
CMD ["/usr/bin/supervisord"]
```

```
[supervisord]
nodaemon=true

[program:tomcat]
command=/opt/tomcat/latest/bin/catalina.sh run
autorestart=true
startsecs=20
stopsignal=INT
stopasgroup=true
killasgroup=true
redirect_stderr=true
stdout_logfile=/var/log/catalina.out
stderr_logfile=/var/log/catalina.out
environment=JAVA_HOME="/usr/lib/jvm/default-jvm",JAVA_BIN="/usr/lib/jvm/default-jvm/bin"

[program:postgres]
command=/usr/local/bin/postgres_custom.sh
autostart=true
autorestart=true
stderr_logfile=/var/log/postgres_err.log
stdout_logfile=/var/log/postgres_out.log
redirect_stderr=true
stopsignal=QUIT
```

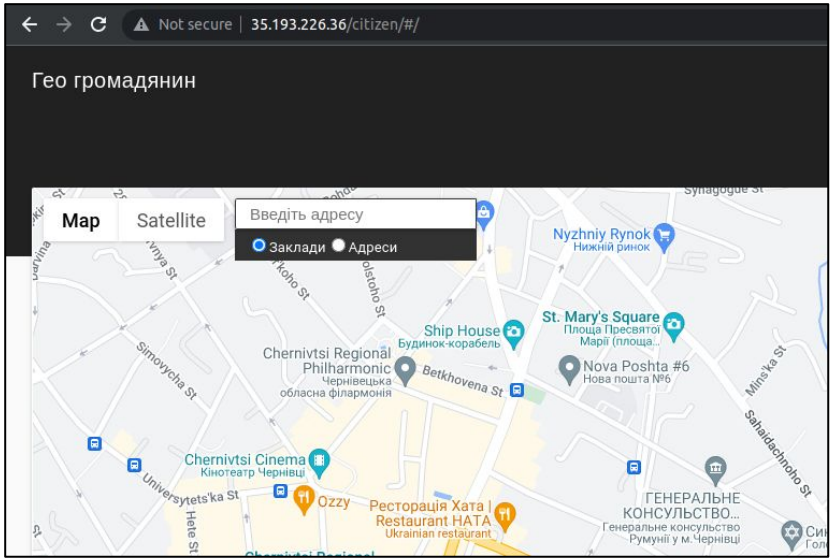
supervisord.conf

softserve

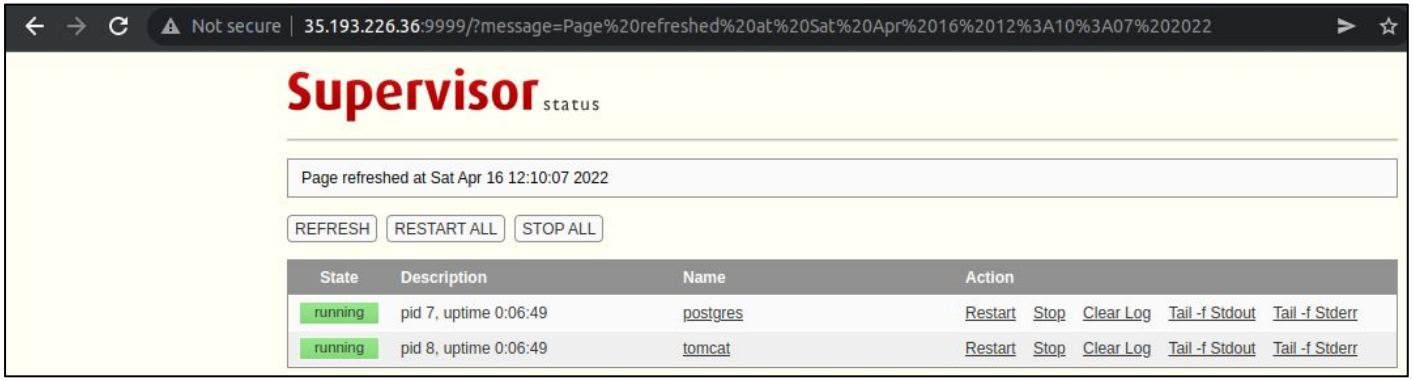
Supervisor: interfaces

Result:

- Geo Citizen page
- Supervisorctl in container
- Supervisor web interface




```
supervisor> status
postgres                         RUNNING   pid 8, uptime 0:29:07
tomcat                           RUNNING   pid 9, uptime 0:29:07
```



Ansible Molecule

Testing of Ansible Roles - DB Geo Citizen:

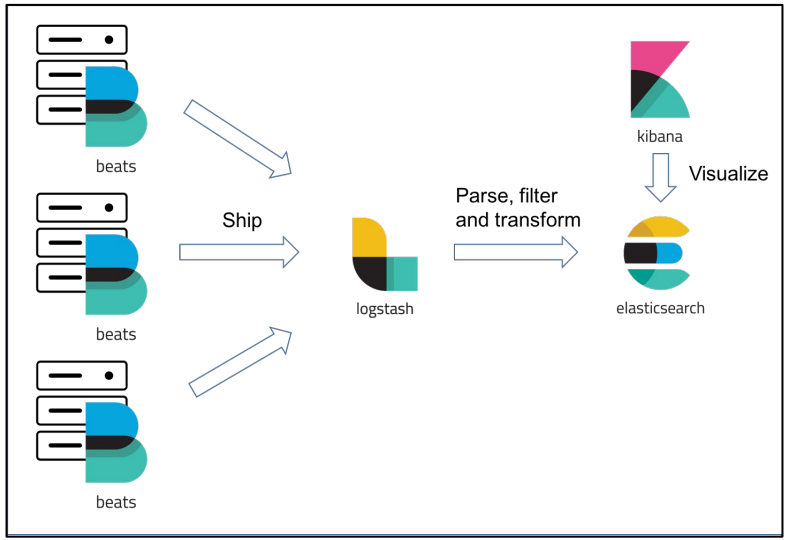
- lint
- simple precheck

```
nce, side_effect, verify, cleanup, destroy
INFO     Performing prerun...
INFO     Set ANSIBLE_LIBRARY=/home/wlados/.cache/ansible-compat/3b4d0d/modules:/home/wlados/.ansible/plugins/modules:/
usr/share/ansible/plugins/modules
INFO     Set ANSIBLE_COLLECTIONS_PATH=/home/wlados/.cache/ansible-compat/3b4d0d/collections:/home/wlados/.ansible/coll
ections:/usr/share/ansible/collections
INFO     Set ANSIBLE_ROLES_PATH=/home/wlados/.cache/ansible-compat/3b4d0d/roles:/home/wlados/.ansible/roles:/usr/share
/ansible/roles:/etc/ansible/roles
INFO     Using /home/wlados/.ansible/roles/wladosnull.ssita_geocitizen_db symlink to current repository in order to en
able Ansible to find the role using its expected full name.
INFO     Running default > dependency
WARNING  Skipping, missing the requirements file.
WARNING  Skipping, missing the requirements file.
INFO     Running default > lint
WARNING: PATH altered to include /usr/bin
WARNING  Loading custom .yamllint config file, this extends our internal yamllint config.
INFO     Running default > cleanup
WARNING  Skipping, cleanup playbook not configured.
INFO     Running default > destroy

PLAY [Destroy] *****************************************************************

TASK [Populate instance config] ************************************************
ok: [localhost]

TASK [Dump instance config] ****************************************************
skipping: [localhost]

PLAY RECAP *********************************************************************
localhost                  : ok=1    changed=0    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

INFO     Running default > syntax

playbook: /home/wlados/Documents/SSITA/Milestone2/galaxy-roles/SSITA_geocitizen_DB/molecule/default/converge.yml
INFO     Running default > create

PLAY [Create] ******************************************************************

TASK [Populate instance config dict] *******************************************
skipping: [localhost]

TASK [Convert instance config dict to a list] **********************************
skipping: [localhost]

TASK [Dump instance config] ****************************************************
skipping: [localhost]

PLAY RECAP *********************************************************************
localhost                  : ok=0    changed=0    unreachable=0    failed=0    skipped=3    rescued=0    ignored=0

INFO     Running default > prepare
WARNING  Skipping, prepare playbook not configured.
INFO     Running default > converge

PLAY [Converge] ****************************************************************
```

Elastic Stack

ELK:

- ElasticSearch
- LogStash
- Kibana



Agents:

- FileBeat
- MetricBeat

Reliably and securely take data from any source, in any format, then search, analyze, and visualize it in real time

softserve

GitLab: custom server

Custom GitLab instance:

- GCP instance
- GitLab
- NOIP
- LE

Additional configs:

- Add SSH key (git clone, push ...)
- Disable registration for anyone
- Create bunch of runners & executors

softserve

GitLab: CI/CD for Supervisor

CI: (developer -> GitHub artefact)

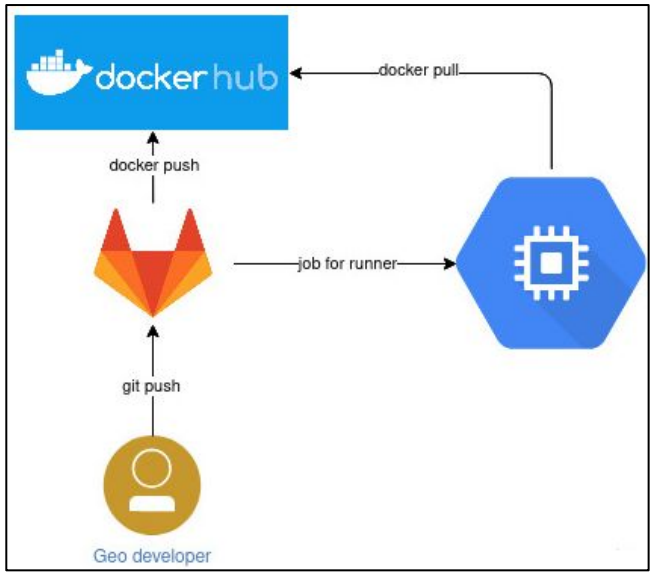
- Write code
- Git ‘push’
- Start pipeline:
  - Fix some values (hosts, creds, paths ...)
  - Build .war artefact
  - Build image

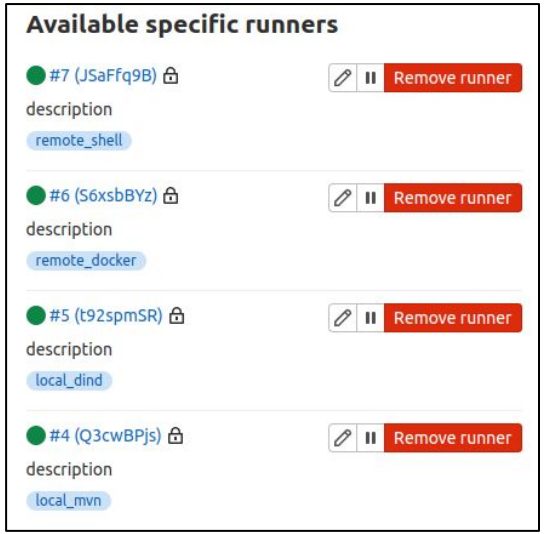
CD: (GitHub artefact -> Docker Hub image)

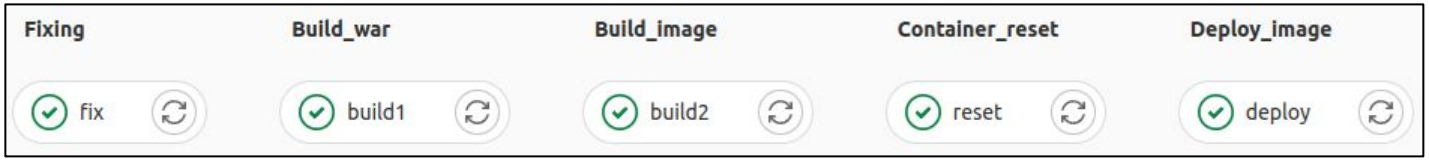
- ...
  - Upload image to private Docker Hub

CD: (Docker Hub image -> Stage server)

- ...
  - Kill available running container
  - Deploy new







Any questions?