System Design of Job scheduler in Golang

System Design - Intro

Goal setting: design a microservices system work that periodically crawls job information, retry if failed. Then do the comparison between user preference and send out information through email.

Codebase: https://github.com/HarrisonLL/Distributed_Job_Crawler

Design choice:

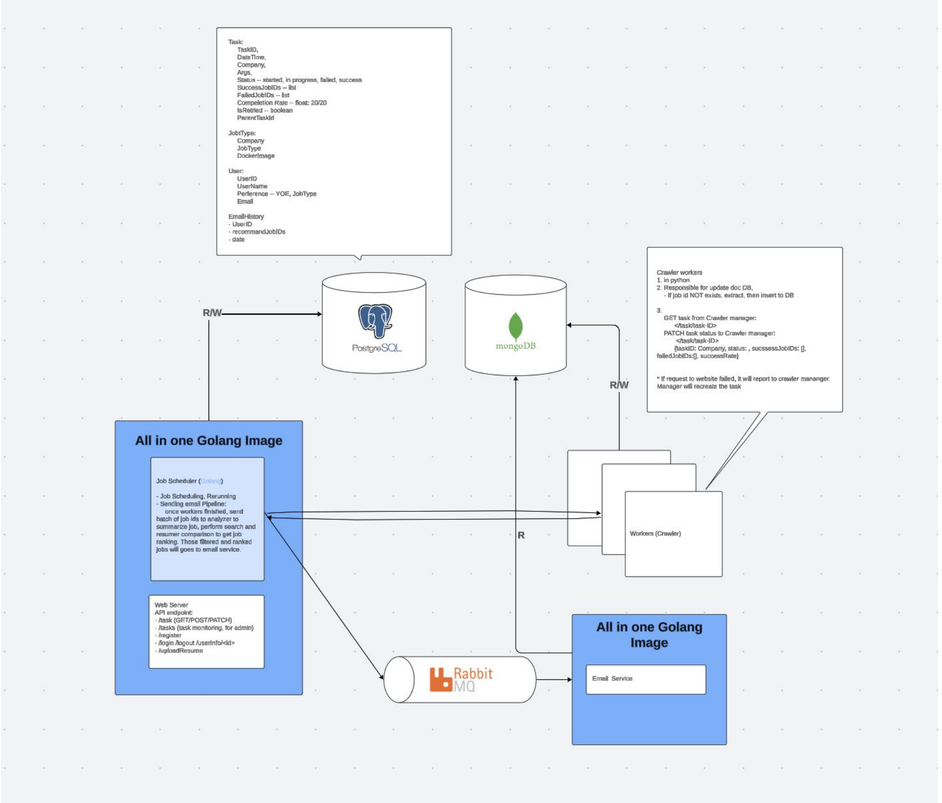
Golang for job scheduler manager, because it is designed for high concurrent applications, and easy to communicate with kubernetes services.

RabbitMQ for emailing queuing, and not used for queuing tasks. Since only the design only targets on spawn and monitor < 10 individual jobs, parallel processes are fine if there are enough CPU cores. However, when numbers of users increase, say hundreds to thousands user, queueing system is needed. This project simulate this situation.

Python for crawler. Python has build in easy-to-use crawling and html parser modules like selumni and beautiful soup.

Docker for containerization. I separated components into different docker containers, which can be managed by kubernetes or docker-compose.

System Design



Task:
TaskID,
DateTime,
Company,
Args,
Status -- started, in progress, failed, success
SuccessJobIDs -- list
FailedJobIDs -- list
Compeletion Rate -- float: 20/20
IsRetried -- boolean
ParentTaskId

JobtType:
Company
JobType
DockerImage

User:
UserID
UserName
Perference -- YOE, JobType
Email

EmailHistory
- UserID
- recommandJobIDs
- date

Crawler workers
1. in python
2. Responsible for update doc DB,
- If job id NOT exists, extract, then insert to DB

3.
GET task from Crawler manager:
</task/task-ID>
PATCH task status to Crawler manager:
</task/task-ID>
{taskID: Company, status: , sucssessJobIDs: [], failedJobIDs:[], successRate}

* If request to website failed, it will report to crawler mananger. Manager will recreate the task

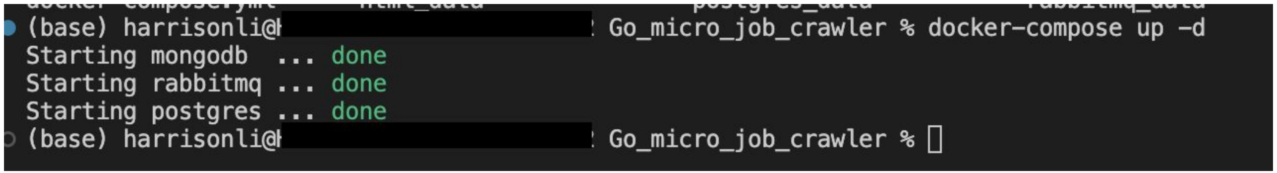
Job Scheduler (Golang)

- Job Scheduling, Rerunning
- Sending email Pipeline:
once workers finished, send batch of job ids to analyzer to summarize job, perform search and resumer comparison to get job ranking. Those filtered and ranked jobs will goes to email service.

Web Server
API endpoint:
- /task (GET/POST/PATCH)
- /tasks (task monitoring, for admin)
- /register
- /login /logout /userInfo/<id>
- /uploadResume

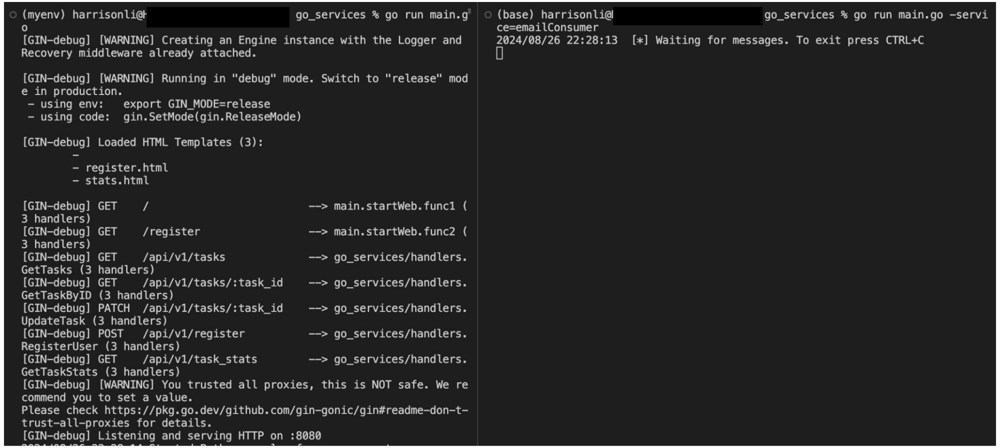
Demo

Step 1. docker-compose up middleware



```
docker-compose.yml      html_data      postgres_data      rabbitmq_data
(base) harrisonli@[redacted] Go_micro_job_crawler % docker-compose up -d
Starting mongodb  ... done
Starting rabbitmq ... done
Starting postgres ... done
(base) harrisonli@[redacted] Go_micro_job_crawler % 
```

Step 2: start two process. Also runnable through docker



```
(myenv) harrisonli@[redacted] go_services % go run main.g
o
[GIN-debug] [WARNING] Creating an Engine instance with the Logger and
Recovery middleware already attached.

[GIN-debug] [WARNING] Running in "debug" mode. Switch to "release" mod
e in production.
 - using env:   export GIN_MODE=release
 - using code:  gin.SetMode(gin.ReleaseMode)

[GIN-debug] Loaded HTML Templates (3):
        -
        - register.html
        - stats.html

[GIN-debug] GET    /                         --> main.startWeb.func1 (
3 handlers)
[GIN-debug] GET    /register                 --> main.startWeb.func2 (
3 handlers)
[GIN-debug] GET    /api/v1/tasks             --> go_services/handlers.
GetTasks (3 handlers)
[GIN-debug] GET    /api/v1/tasks/:task_id    --> go_services/handlers.
GetTaskByID (3 handlers)
[GIN-debug] PATCH  /api/v1/tasks/:task_id    --> go_services/handlers.
UpdateTask (3 handlers)
[GIN-debug] POST   /api/v1/register          --> go_services/handlers.
RegisterUser (3 handlers)
[GIN-debug] GET    /api/v1/task_stats        --> go_services/handlers.
GetTaskStats (3 handlers)
[GIN-debug] [WARNING] You trusted all proxies, this is NOT safe. We re
commend you to set a value.
Please check https://pkg.go.dev/github.com/gin-gonic/gin#readme-don-t-
trust-all-proxies for details.
[GIN-debug] Listening and serving HTTP on :8080
```

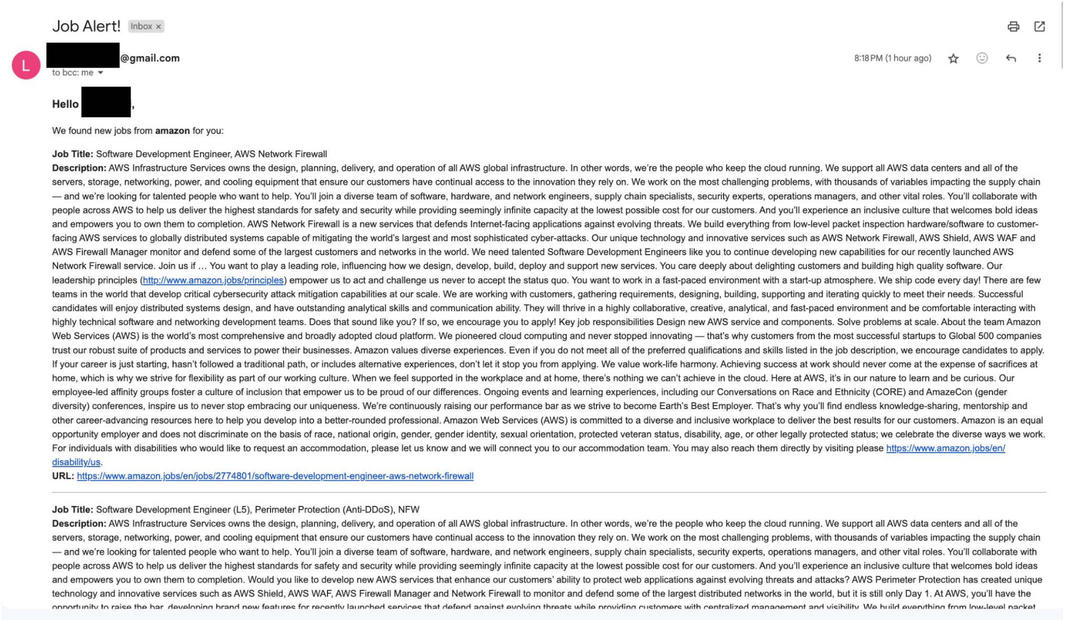
```
(base) harrisonli@[redacted] go_services % go run main.go -servi
ce=emailConsumer
2024/08/26 22:28:13  [*] Waiting for messages. To exit press CTRL+C
```

Demo

Steps 3:

Jobs is scheduled every 6 hours, retry happens in hour of the initial job. Once job is found it will send emails to users.

Sample email:



Job Alert!

[redacted]@gmail.com
to bcc: me

Hello [redacted]**,**

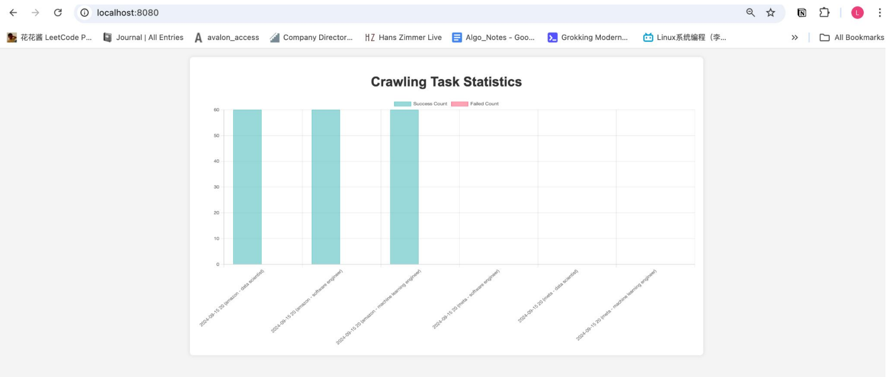
We found new jobs from **amazon** for you:

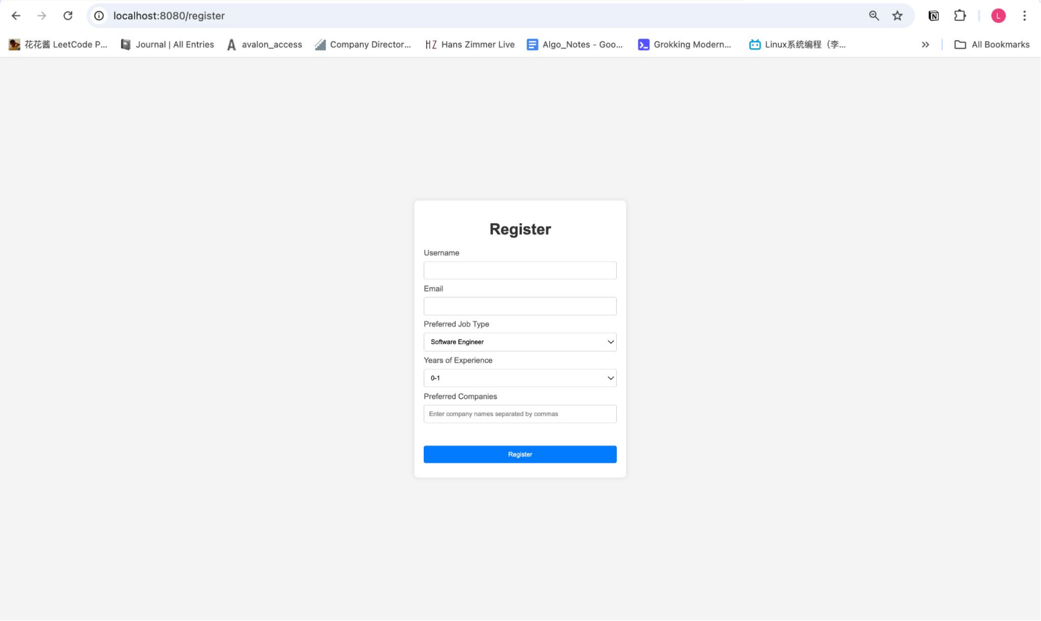
Job Title: Software Development Engineer, AWS Network Firewall
Description: AWS Infrastructure Services owns the design, planning, delivery, and operation of all AWS global infrastructure. In other words, we're the people who keep the cloud running. We support all AWS data centers and all of the servers, storage, networking, power, and cooling equipment that ensure our customers have continual access to the innovation they rely on. We work on the most challenging problems, with thousands of variables impacting the supply chain — and we're looking for talented people who want to help. You'll join a diverse team of software, hardware, and network engineers, supply chain specialists, security experts, operations managers, and other vital roles. You'll collaborate with people across AWS to help us deliver the highest standards for safety and security while providing seemingly infinite capacity at the lowest possible cost for our customers. And you'll experience an inclusive culture that welcomes bold ideas and empowers you to own them to completion. AWS Network Firewall is a new services that defends Internet-facing applications against evolving threats. We build everything from low-level packet inspection hardware/software to customer-facing AWS services to globally distributed systems capable of mitigating the world's largest and most sophisticated cyber-attacks. Our unique technology and innovative services such as AWS Network Firewall, AWS Shield, AWS WAF and AWS Firewall Manager monitor and defend some of the largest customers and networks in the world. We need talented Software Development Engineers like you to continue developing new capabilities for our recently launched AWS Network Firewall service. Join us if … You want to play a leading role, influencing how we design, develop, build, deploy and support new services. You care deeply about delighting customers and building high quality software. Our leadership principles (http://www.amazon.jobs/principles) empower us to act and challenge us never to accept the status quo. You want to work in a fast-paced environment with a start-up atmosphere. We ship code every day! There are few teams in the world that develop critical cybersecurity attack mitigation capabilities at our scale. We are working with customers, gathering requirements, designing, building, supporting and iterating quickly to meet their needs. Successful candidates will enjoy distributed systems design, and have outstanding analytical skills and communication ability. They will thrive in a highly collaborative, creative, analytical, and fast-paced environment and be comfortable interacting with highly technical software and networking development teams. Does that sound like you? If so, we encourage you to apply! Key job responsibilities Design new AWS service and components. Solve problems at scale. About the team Amazon Web Services (AWS) is the world's most comprehensive and broadly adopted cloud platform. We pioneered cloud computing and never stopped innovating — that's why customers from the most successful startups to Global 500 companies trust our robust suite of products and services to power their businesses. Amazon values diverse experiences. Even if you do not meet all of the preferred qualifications and skills listed in the job description, we encourage candidates to apply. If your career is just starting, hasn't followed a traditional path, or includes alternative experiences, don't let it stop you from applying. We value work-life harmony. Achieving success at work should never come at the expense of sacrifices at home, which is why we strive for flexibility as part of our working culture. When we feel supported in the workplace and at home, there's nothing we can't achieve in the cloud. Here at AWS, it's in our nature to learn and be curious. Our employee-led affinity groups foster a culture of inclusion that empower us to be proud of our differences. Ongoing events and learning experiences, including our Conversations on Race and Ethnicity (CORE) and AmazeCon (gender diversity) conferences, inspire us to never stop embracing our uniqueness. We're continuously raising our performance bar as we strive to become Earth's Best Employer. That's why you'll find endless knowledge-sharing, mentorship and other career-advancing resources here to help you develop into a better-rounded professional. Amazon Web Services (AWS) is committed to a diverse and inclusive workplace to deliver the best results for our customers. Amazon is an equal opportunity employer and does not discriminate on the basis of race, national origin, gender, gender identity, sexual orientation, protected veteran status, disability, age, or other legally protected status; we celebrate the diverse ways we work. For individuals with disabilities who would like to request an accommodation, please let us know and we will connect you to our accommodation team. You may also reach them directly by visiting please https://www.amazon.jobs/en/disability/us.
URL: https://www.amazon.jobs/en/jobs/2774801/software-development-engineer-aws-network-firewall

Job Title: Software Development Engineer (L5), Perimeter Protection (Anti-DDoS), NFW
Description: AWS Infrastructure Services owns the design, planning, delivery, and operation of all AWS global infrastructure. In other words, we're the people who keep the cloud running. We support all AWS data centers and all of the servers, storage, networking, power, and cooling equipment that ensure our customers have continual access to the innovation they rely on. We work on the most challenging problems, with thousands of variables impacting the supply chain — and we're looking for talented people who want to help. You'll join a diverse team of software, hardware, and network engineers, supply chain specialists, security experts, operations managers, and other vital roles. You'll collaborate with people across AWS to help us deliver the highest standards for safety and security while providing seemingly infinite capacity at the lowest possible cost for our customers. And you'll experience an inclusive culture that welcomes bold ideas and empowers you to own them to completion. Would you like to develop new AWS services that enhance our customers' ability to protect web applications against evolving threats and attacks? AWS Perimeter Protection has created unique technology and innovative services such as AWS Shield, AWS WAF, AWS Firewall Manager and Network Firewall to monitor and defend some of the largest distributed networks in the world, but it is still only Day 1. At AWS, you'll have the opportunity to raise the bar, developing brand new features for recently launched services that defend against evolving threats while providing customers with centralized management and visibility. We build everything from low-level packet

Demo

Stats and register UI:


localhost:8080
花花酱 LeetCode P...
Journal | All Entries
avalon_access
Company Director...
Hans Zimmer Live
Algo_Notes - Goo...
Grokking Modern...
Linux系统编程（李...
All Bookmarks
Crawling Task Statistics
Success Count
Failed Count
60
50
40
30
20
10
0
2024-09-15 20 (amazon - data scientist)
2024-09-15 20 (amazon - software engineer)
2024-09-15 20 (amazon - machine learning engineer)
2024-09-15 20 (meta - software engineer)
2024-09-15 20 (meta - data scientist)
2024-09-15 20 (meta - machine learning engineer)



Register

Username

Email

Preferred Job Type: Software Engineer

Years of Experience: 0-1

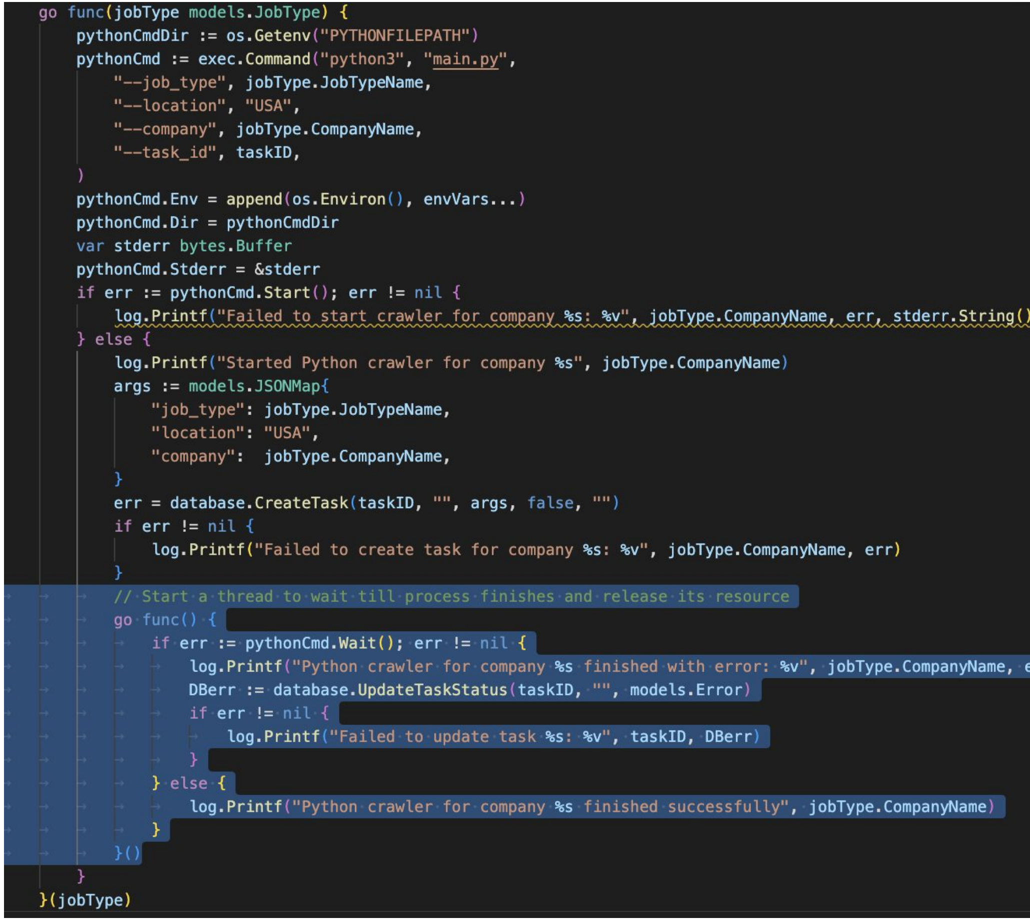
Preferred Companies: Enter company names separated by commas

Register

Study notes:

Since I am very new to GoLang, I have learned a lot from doing this project.

1. GoRoutine: go's simple way of doing multithreading
2. GoChannel: go's way to sync threads. We can also run as infinite for loop to listen to something from message queue. (refer: https://www.rabbitmq.com/tutorials/tutorial-one-go)
3. Other than that, I have also experimented some system programming. For example, the code on the right shows I have started many threads to run a python program, for each thread, I also started additional thread to wait till the program finishes and release its resource. (refer: https://pkg.go.dev/os/exec)



```go
go func(jobType models.JobType) {
    pythonCmdDir := os.Getenv("PYTHONFILEPATH")
    pythonCmd := exec.Command("python3", "main.py",
        "--job_type", jobType.JobTypeName,
        "--location", "USA",
        "--company", jobType.CompanyName,
        "--task_id", taskID,
    )
    pythonCmd.Env = append(os.Environ(), envVars...)
    pythonCmd.Dir = pythonCmdDir
    var stderr bytes.Buffer
    pythonCmd.Stderr = &stderr
    if err := pythonCmd.Start(); err != nil {
        log.Printf("Failed to start crawler for company %s: %v", jobType.CompanyName, err, stderr.String()
    } else {
        log.Printf("Started Python crawler for company %s", jobType.CompanyName)
        args := models.JSONMap{
            "job_type": jobType.JobTypeName,
            "location": "USA",
            "company":  jobType.CompanyName,
        }
        err = database.CreateTask(taskID, "", args, false, "")
        if err != nil {
            log.Printf("Failed to create task for company %s: %v", jobType.CompanyName, err)
        }
        // Start a thread to wait till process finishes and release its resource
        go func() {
            if err := pythonCmd.Wait(); err != nil {
                log.Printf("Python crawler for company %s finished with error: %v", jobType.CompanyName, e
                DBerr := database.UpdateTaskStatus(taskID, "", models.Error)
                if err != nil {
                    log.Printf("Failed to update task %s: %v", taskID, DBerr)
                }
            } else {
                log.Printf("Python crawler for company %s finished successfully", jobType.CompanyName)
            }
        }()
    }
}(jobType)
```