

## 基本情報

フリガナ	チン ギョウケツ
氏名	陳 曉傑
Email	<a href="mailto:caliburn1994@gmail.com">caliburn1994@gmail.com</a>
学歴	2017-06 青島大学（ソフトウェアエンジニアリング）
出生地	中国広東
資格・言語能力	日本語一級
	簡単英会話
	英語の読み書きができます。
最寄駅	西武池袋線一ひばりが丘

## 自己PR

フルスタックエンジニアですが、バックエンドのほうが上手です。フロントエンドはJS、HTML、thymeleafなどができます。バックエンドはJava と Spring はメインです。サーバー側は Bash などができます。K8s、Docker、AWS などが学ぶことがあります。英語読む能力があり、[StackOverflow](#) にはポイントがありますので、解決能力には自信があります。[GitHub](#)でいくつ練習用プロジェクト、Spring Data Rest、Spring MVC、Spring Cloud Config などができます。

## 技術・スキル

**基礎的な collection と algorithm は勉強しましたが、すぐ答えられない。ノート見るまたは復習する必要があります。Leetcodeで練習する習慣はありません。**現在、AWSの基本知識または開発に関することを勉強しています。

## Web

\_ =実務経験有り    ✕ =実践経験無し

技術			備考
<u>Java</u>	version = 7/8/15		
	lambda	<u>optional</u>	
		stream	ノート見る必要
	collection	<u>HashMap</u> 、 <u>ArrayList</u>	
	✕ thread、JVM		
	<u>SpringBoot</u> (version=1/2)		ゼロから作るのは可能。基本使い方はしています。
	Template: <u>Apache Wicket</u> 、 <u>JSP</u> 、JSF		簡単なプログラミング
	Spring thymeleaf		ゼロから作るのは可能
	<u>Maven</u> 、 <u>Gradle</u> ( <u>groovy</u> )		ゼロから作るのは可能
	Rest: Spring Data Rest (hateoas)		ゼロから作るのは可能
	ORM: <u>MyBatis</u> 、Spring Data Rest (hibernate)		ゼロから作るのは可能
	Cloud: Spring Cloud Config		ゼロから作るのは可能
<u>python</u>	version=3.7		
	<u>Django</u>		ゼロから作るのは可能
	✕ thread		
<u>JavaScript</u>	Ajax、jQuery		簡単なプログラミング、
Go			簡単なツールを作ったことがあります、そして興味があります。
	✕ thread、framework		

## DB

\_ =実務経験有り    ✕ =実践経験無し

技術	備考
<u>MySQL</u>	B+Treeの基本知識
	index merge、composite key
	UUIDとauto increment
	✕ 事務（Transaction）の(事务隔离)Isolation
<u>PostgreSQL</u>	sequence
MSSQL	大学のレッスンで

## Deploy&&Backend

\_ =実務経験有り    ✕ =実践経験無し

技術			備考
Language	<u>Bash</u>	pipe、getopts、systemd、cron	
		SSH、curl	
OS	<u>Windows</u>		
	<u>Linux</u>	Ubuntu Desktop	基本知識が持つ
		<u>Centos</u>	操作が可能
<u>Docker</u>	Dockerfile		
	Docker Compose		
	<a href="#">Testcontainers</a> ( Java )		
<u>Vagrant</u>			
Kubernetes	helm 3		
	minikube		
AWS	LocalStack		
	EC2		
	S3		
	EKS		
	✕ CloudFormation、Terraform		
<u>Jenkins</u>	<u>Jenkins + Maven</u>		

## Other

\_ =実務経験有り    ✕ =実践経験無し

技術
<u>Jetbrains</u>
<u>Tera Term</u>
<u>GitLab、Redmine</u>

## 職務経歴

### 2017.7～2018.8 - Shanghai Koal Software Co 上海格尔

事業内容：主に公開鍵基盤（PKI）に基づく商用パスワードソフトウェア製品の研究開発、製造、販売を行う。

従業員数： 300人

#### 電子証明書の管理システム

[プロジェクト概要]

電子証明書発行の一環とする管理システムです。端末の接続の設定または環境設定を管理するシステムです。

[規模]

要員10名

[開発環境]

- OS: windows, centos
- DB: MySQL 5.6
- 言語: Java 7
- フレームワーク: Apache Wicket
- その他: Maven, Gradle

[担当]

詳細設計・プログラミング／単体テスト （役割：メンバー）

[業務内容]

- 機能実装
- MySQL移行

[習得技術]

- Gradleの最初の接触
- Wicketの使い方

## 2019.2～2020.2 - テラテクノロジー株式会社（派遣先）

### 端末管理システム

#### [プロジェクト概要]

セットトップボックスなどの端末の情報を管理するシステム。Javaで端末からのメッセージを受けて、BashでDB操作含める全ての内容を行います。

#### [規模]

要員10名

#### [開発環境]

- OS: centos6
- 言語: bash
- DB: Postgresql

#### [担当]

プログラミング／単体・結合テスト（役割：メンバー）

#### [業務内容]

- Bashで機能を実装
- VMを使ってテスト

#### [習得技術]

- Linux及びBashのいろんな使い方

### 管理システムとツール

#### [プロジェクト概要]

Rubyで構築されていたプロジェクトは、Pythonで書き直し、そして、機能追加。

#### [規模]

要員1名

#### [開発環境]

- OS: Centos
- 言語: python 3, JavaScript
- フレームワーク: Django

#### [担当]

プログラミング／結合・総合テスト

#### [業務内容]

1. Rubyで構築されていたWeb Mvcシステムを、PythonのDjangoでもう一度実装して、そして新たな機能を追加します。
2. 他のシステムに必要なツールを作ります。ハートビート(heartbeat)のような機能

#### [習得技術]

- PythonとDjangoの使い方
- Linuxで、Pythonの複数バージョンのやり方
- Vagrantの使い方

## 2020.3～2020.8

Docker、kubernetes、AWS、GKE（Google Kubernetes Engine）、Linuxなどを独学していました。技術を深く理解するために、英語も独学していました。時に、資料を通訳したり、Stackoverflowで質問したり、答えたりしています。

## 2020.9～ 日本恒生軟件（本社）

事業内容：決済サービス事業、派遣事業など。

### 決済のゲットウェイシステム

[プロジェクト概要]

あらゆるスマホ決済をOneアプリで実現するサービス。

[規模]

要員10-20名

[開発環境]

- OS: window, linux
- 言語: Java 8
- DB: MySQL5.6
- フレームワーク: Spring Boot, MyBatis
- その他: Maven, Jenkin, AWS

[担当]

運用・保守・機能追加（役割：メンバー）

[業務内容]

- テスト環境を構築、テストコードを追加
- 機能追加
- MySQLテーブルの設計それとパフォーマンスついてのアドレス

[習得技術]

- MySQLテーブルの設計、及びIndexの効率改善
- Spring構築の知識
- Springのテストの書き方、自動化テストのやり方
- Dockerでのテストコード

## 他に

---

### ブログ

Jekyllで生成の[ブログ](#)。

### 練習用プロジェクト

- [Spring-Data-REST-Demo](#) : Restサービス及びSpring Dataの使い方
- [Spring-Cloud-Config-Demo](#) : 配置ファイルを管理するサーバーの作り方
- [SpringBootNote](#): MyBatisとMVCの使い方、それと、Gradleの管理方法の練習
- [ubuntu-minikube](#) : ローカルクラウドの作り方、開発に便利な環境の作り方

### Stackoverflow

現在 [Stackoverflow](#) の信用度は800以上。

