

# Using Awesome-CV to Generate a Professional Résumé

## Introduction

Before using ZhiyeApp, I switched to the powerful and customizable Awesome-CV. This LaTeX-based template makes creating professional résumés easy and highly customizable.

---

## Why Awesome-CV?

- **Customizable:** You can personalize sections, colors, and formatting.
  - **Professional Look:** Clean design perfect for job applications.
  - **Easy to Use:** Requires minimal LaTeX knowledge.
- 

## My Résumé Example

Here is a simplified version of the `resume.tex` file I use:

```
%-----  
% CONFIGURATIONS  
%-----  
\documentclass[11pt, a4paper]{awesome-cv}  
  
% Page margins and section highlights  
\geometry{left=1.4cm, top=.8cm, right=1.4cm, bottom=1.8cm, footskip=.5cm}  
\colorlet{awesome}{awesome-red}  
\setbool{acvSectionColorHighlight}{true}  
  
%-----  
% PERSONAL INFORMATION  
%-----  
\name{Zhiwei}{Li}  
\position{Full Stack Engineer{\enskip\cdotp\enskip}Backend Engineer}  
\address{Guangzhou, China}  
\mobile{(+86) 132-6163-0925}  
\email{lzwjava@gmail.com}
```

```

\homepage{https://lzwjava.github.io}
\github{lzwjava}
\linkedin{lzwjava}
\quote{``Freedom Truth"}

%-----
\begin{document}

% Header and Footer
\makecvheader[C]
\makecvfooter{\today}{Zhiwei Li~~~ · ~~~Résumé}{\thepage}

% Content Sections
\input{resume/summary.tex}
\input{resume/experience.tex}
\input{resume/education.tex}
\input{resume/corporateprojects.tex}
\input{resume/personalprojects.tex}
\input{resume/blogposts.tex}
\input{resume/papers.tex}
\input{resume/books.tex}
\input{resume/skills.tex}
\input{resume/tools.tex}
\input{resume/knowledge.tex}
\input{resume/certificates.tex}

\end{document}

```

Makefile for Automation

To automate the PDF generation process, I use the following Makefile:

```

```Makefile
.PHONY: awesome-cv

CC = xelatex

```

```

EXAMPLES_DIR = awesome-cv
RESUME_DIR = awesome-cv/resume
RESUME_ZH_DIR = awesome-cv/resume-zh
RESUME_SRCS = $(shell find $(RESUME_DIR) -name '*.tex')
RESUME_ZH_SRCS = $(shell find $(RESUME_ZH_DIR) -name '*.tex')

awesome-cv: $(foreach x, coverletter resume-zh resume, $x.pdf)

resume.pdf: $(EXAMPLES_DIR)/resume.tex $(RESUME_SRCS)
    $(CC) -output-directory=$(EXAMPLES_DIR) $<

resume-zh.pdf: $(EXAMPLES_DIR)/resume-zh.tex $(RESUME_ZH_SRCS)
    $(CC) -output-directory=$(EXAMPLES_DIR) $<

coverletter.pdf: $(EXAMPLES_DIR)/coverletter.tex
    $(CC) -output-directory=$(EXAMPLES_DIR) $<

clean:
    rm -rf $(EXAMPLES_DIR)/*.pdf

```

## How It Works

### 1. Generate PDFs:

- Run `make awesome-cv` to generate the following PDF files:
  - `resume.pdf`: English résumé
  - `resume-zh.pdf`: Chinese résumé
  - `coverletter.pdf`: Cover letter

### 2. Clean Up:

- Run `make clean` to remove all generated PDF files.

## Conclusion

By leveraging Awesome-CV and this Makefile setup, generating and maintaining professional résumés becomes effortless. Whether you're applying for technical roles or sharing your achievements, Awesome-CV helps you present your work beautifully and efficiently.

Check out the Awesome-CV repository for more details: [Awesome-CV on GitHub](#).