

CPD Stage 2 - URL Shortener Service using Go

Zarraf Khan - 45949344

Definition of Done - What is my project?

Essentially, my CPD will involve creating a URL Shortening service, as a result, using these technologies. This will be in the form of a web application. As deployment is not my current focus, it will remain as a self-hosted project for now, available on a repository. If time permits, I will deploy it.

Front End

- The final project should consist of 2 pages (Main, Error).
- The main page should show of a textarea, where users can input a single, or a set of links to shorten
- After the user submits the links, it should immediately respond with a shortened link which redirects anyone to the inputs
- A QR code would appear adjacent to the link, simarly encoded to that particular input (**Not a MVP**)
- Any invalid inputs, or caching failiures would redirect the user into a temporary error page
- I will most likely use Tailwind CSS to write this portion

Back End

- The majority of this project consists of learning Go, and familiarising myself with frameworks such as MongoDB
- Specifically, this includes how to appropriately structure and approach development
- The back end will most likely consist of seperate modules (url handling, data caching and error handling)
- I expect to write some simple tests for these modules
- For now, there will not be any User login or authentication

Learning Goals

Main Goal : To gain a strong competency in Go for backend development.

Sub Goals :

- Learn how to create programs in Go
- Learn how to design and build REST API
- Learn how to use frameworks like MongoDB to cache data and scale apps (instead of using preset CSV or JSON file)
- Reinforce how to make responsive UI (time sensitive)

I would like to gain more exposure into backend technologies and API building. As such, I would like to learn Golang (Go) as a starting point, and aim to further utilise this as a daily driver. I have limited experience using DB management systems, so similarly, I would like to be able to gain further exposure in the field.

In addition, this is not a concerning goal in this project, but I would like to learn how to properly implement responsive UI using CSS libraries. I have some experience in this, but I want to further develop these skills.

Main Learning Objectives

By the end of my CPD, I expect to be able to:

- Apply Go as a daily driver in future projects
- Demonstrate the ability to use Go to create and deploy CLI tools, as well as APIs
- Apply knowledge on concurrency through Goroutines in the future development
- Understand basic usage of handling HTTP requests (using Fiber framework), and apply it to developing the API
- Apply knowledge on data storage and caching using frameworks such as MongoDB in the deployment of the application (self-hosted)
- Apply CSS libraries to my responsive, front end projects

Expected Timeline

Weeks	Goals	Deliverables*
Week 1	Learn basic Go syntax	-
Week 2	Learn Goroutines (concurrency), channels, generics	Write basic program/s which incorporates learning outcomes (including week 1), Setup project structure

Weeks	Goals	Deliverables*
Week 3	Learn REST API building (middleware, authentication, package management)	Write simple HTTP handler function which redirects user to preset links (CLI tool)
Week 4	Learn Fibre framework for front-end development and MongoDB for caching	Set up MongoDB cluster, and get list of DBs, incorporate live DB in URL Shortner Service (CLI)
Week 5-6	Create simple URL shortener service (simple front-end)**	Continue development on web application, Add further complexities/features (if time permits)

*as part of my weekly learning, all progress (exercises done) should be documented on the portfolio repo

Measures of Success

*(in order of complexity)

- To be able to write simple programs in Go
- To have a working CLI tool which takes uses an in-memory list, shortnens the links, and has the new links redirect the user to the original one
- To incoporate a DB to cache links and return shortened version
- To redirect it to multiple links (single shortened URL)
- To have a simple front-end (proper web app)
- To have basic QR Code embedding or other additional features

Off-Ramps & Potential Risks

For each week, it would be required to document and present my learned materials on my portfolio. The main deliverables would include this, along with the application.

- The main risk factor I suspect would be spending too much time on the handler logic in week 3 - the bulk of the link shortner logic should be developed by week 4
- Week 4 - API without front-end (CLI based), or other complexities such as embedding link into QR code or user Auth