Summary Report

CST8333 19F

Professor: Stanley Pieda

By: Lucas Estienne (esti0011)

040 819 959

## What was learned about my programming language

I picked Go as my programming language to learn this semester.

The topics I learned and implemented include unit testing, simple data structures, more advanced ones, file I/O, loops, reflection, string manipulation, input validation, threading, goroutines, waitgroups, and database drivers.

Golang’s biggest strength is arguably its non-blocking concurrency, and is really Go’s major selling point. It’s got garbage collection, fast compilation, fast execution, an awesome standard library and documentation, generally great tooling, very high performance while maintaining a low memory footbrint, and an easy to learn syntax. It doesn’t require a VM like say Java.

However, some of its weaknesses include no compile-time memory safety, no generics, generally large binaries, poor support for things like GUIs, very verbose error handling, no immutability, and no zero-value initialization.

## The Best Resources for Me to Learn Are

|  |  |  |
| --- | --- | --- |
| **Resource** | **Time Consumption** | **Usefulness** |
| Official documentation | Medium | High |
| Tutorials | Medium | Medium |
| Articles | Low | Medium |

Those are the only resources I used. The best resource for me was for sure Go’s official documentation, because it’s very informative and concise, and easily to reference.

## Learning Process Phases and Time Taken

From most time consuming, to least time consuming:

* 1. Planning
  2. Research
  3. Basic Topics
  4. Advanced Topics
  5. Project Delivery

Planning & research, which not programming tasks, took FAR MORE time than the other phases, due to all the “boilerplate” like making a WBS and writing down a bunch of learning resources that to be honest weren’t that useful at all. Actually programming took basically no time at all, and to be honest I probably spent more time on Assignment 1, 2 and Exercise 2 than on Assignment 3,4 and the final project, and all the programming exercises combined…

## WBS, Project Management Software, Reflection on Time Estimation

I mean, it was completely inaccurate for the simple reason that I just did the WBS as a “bare minimum” for the requirements of the course. I already know how to do estimation, I’ve been doing it professionally for the past 4 years now almost, and as the scrum master for my team at work for the past year or so. This part of the course was honestly a waste of time for me. The values I put in the WBS I knew from the start I wouldn’t stick to, and to be honest I spent less than 3 hours on the entirety of Assignment 3, less than 2 hours on Assignment 4 and about an hour and a half on the Final Project.

## Discussion Board Post Archive

## Hybrid 1

**1. In the past week(s) you have learned(i.e., researched, read, applied) new aspects of your chosen programming language. What did you learn about your programming language?**

I've learned about the Go basics, unit testing, proper code style, and Go idioms. I've also set up my environment (using GVM), and editor (easy, vim for life baby)

Go is an extremely powerful language, which is getting quite a bit of traction thanks to the explosion of popularity with technologies like Terraform, Kubernetes, etc.

**2. With regard to the learning resources you used to learn: What one worked best, and which one was less successful and why?**

I tend to default to the official Go documentation, which is regarded as one of the best among languages for it's conciseness and in general very useful: <https://golang.org/doc/>

As well as the standard library packages documentation: <https://golang.org/pkg/>

**3. What phase of this course did you just complete as part of your overall learning process?**

I've pretty much completed the research phase this past week, although there will be more research to be done for sure this was the bulk of it!

**4. What did you learn about time estimation? How did your time estimates work out, did you over-estimate or under-estimate your time usage?**

Honestly I've been working full time in an enterprise setting doing development & infrastructure work for the past 3 1/2 years, so I'm pretty comfortable with estimation already. Pretty much nothing learned here.

My estimates were right on track for this first week, I've gotten pretty good at them since I've been my team's Scrum Master for over a year now at work.

## Hybrid 2

**1. In the past week(s) you have learned(i.e., researched, read, applied) new aspects of your chosen programming language. What did you learn about your programming language?**

I've learned about simple data structures, file  I/O, loops, and gained a good grasp of the general flow of programming in Go.

**2. With regard to the learning resources you used to learn: What one worked best, and which one was less successful and why?**

Still pretty much learning from the official Go documentation, doing pretty well so far.

**3. What phase of this course did you just complete as part of your overall learning process?**

I'd say I've completed the planning phase of the project, with Assignment 2 done as well as Exercises 2 & 3.

**4. What did you learn about time estimation? How did your time estimates work out, did you over-estimate or under-estimate your time usage?**

Like I mentioned in the last hybrid, I've been doing estimates for the past 3+ years at work, the last 1 1/2 of which I've been scrum master for my team - so I haven't really learned anything new about estimations.

I'm pretty much on track.

## Hybrid 3

**1. In the past week(s) you have learned(i.e., researched, read, applied) new aspects of your chosen programming language. What did you learn about your programming language?**

I've learned about more file I/O, more complex data structures, struct reflection, complex string manipulation, and writing CLI prompts/validating input in Go. I feel like I have a pretty good grasp so far.

**2. With regard to the learning resources you used to learn: What one worked best, and which one was less successful and why?**

Still the official Go documentation, haven't needed anything else.

**3. What phase of this course did you just complete as part of your overall learning process?**

I'd say I've completed the Minimum Viable Product for this course, with Assignment 3 done. I've got the basics down as well as some advanced concepts

**4. What did you learn about time estimation? How did your time estimates work out, did you over-estimate or under-estimate your time usage?**

Like I mentioned in the last hybrid, I've been doing estimates for the past 3+ years at work, the last 1 1/2 of which I've been scrum master for my team - so I haven't really learned anything new about estimations.

I'm pretty much on track.

## Hybrid 4

**1. In the past week(s) you have learned(i.e., researched, read, applied) new aspects of your chosen programming language. What did you learn about your programming language?**

I've learned about database drivers in Go, as well as multithreading using waitgroups and goroutines. I feel like I've gotten a good grasp on both of those concepts

**2. With regard to the learning resources you used to learn: What one worked best, and which one was less successful and why?**

Still the official Go documentation, and the package documentation for the sqlite3 driver. Haven't needed much else.

**3. What phase of this course did you just complete as part of your overall learning process?**

I've completed the Advanced Topics phase of the course, with Assignment 4 done.

**4. What did you learn about time estimation? How did your time estimates work out, did you over-estimate or under-estimate your time usage?**

Like I mentioned in the last hybrid, I've been doing estimates for the past 3+ years at work, the last 1 1/2 of which I've been scrum master for my team - so I haven't really learned anything new about estimations.

I'm pretty much on track.