

TUIs a final topic





- The project
 - Any questions
- The presentation
 - About 5-7 minutes each.
 - Show us and tell us what you did
 - And how awesome it is.
 - Present well
 - Instructor demonstrates some examples of not so well.

One Last topic: TUIs



- TUI
 - Terminal User Interface
- More than just a command line interface, runs in the terminal and gives you a User Interface
- Lots of stuff runs in the command line on a server somewhere.
- It COULD run in a terminal user interface on the server somewhere.

Go and TUIs



- Go seems like a good match for TUIs,
 - after all lots of go is run on the back end of a server somewhere.
- If we look at the awesome-go list, the list of TUI libraries and helpers is several times as long as the GUI section
- <https://github.com/avelino/awesome-go?tab=readme-ov-file#command-line>
 - Though a few of these are a little older

The two most widely used



- It seems that there are two libraries most widely used for TUIs in go
 - bubbletea: <https://github.com/charmbracelet/bubbletea>
 - A widely used functional TUI library.
 - tview <https://github.com/rivo/tview>
 - A widely used object-oriented-ish TUI library.
 - I'll look at tview with you all, others might look at bubbletea

app



- The first struct we need is app
 - app encapsulates
 - The application run loop
 - Holds the main widget/component
 - Holds on to the app-wide preferences

InputField



- **InputField**
 - Encapsulates what is needed to be a type-able input field
 - Includes the label
 - And optionally sample text
- **Example:**

```
dataDisplay.Company = tview.NewInputField()  
dataDisplay.Company.SetLabel("Company:")  
dataDisplay.Company.SetFieldWidth(30)
```

Form



- The Form struct in tview is used to hold multiple interface elements.
- It might have been called screen in another era
- Example from docs →

```
form := tview.NewForm()  
form.SetBorder(true)  
form.AddFormItem(dataDisplay.Company)  
form.AddFormItem(dataDisplay.JobTitle)  
form.AddFormItem(dataDisplay.Location)
```

Enter some data

Title	Mr.
First name	Oliver
Last name	Rivo
Address	Somewhere in the world 😊
Notes	This is just a demo. You can enter whatever you wish.
Age 18+	X
Password	*****

Save Quit

Checkbox



- Checkbox encapsulates true false data GUI
- There are a lot of other adjustments we can make
- Let's look at a few of the options

```
dataDisplay.IsUSA = tview.NewCheckbox()  
dataDisplay.IsUSA.SetLabel("Is US Job:")
```

```
dataDisplay.IsUSA.setChecked(firstItem[3] == "us")
```

Button



- There is a NewButton function
- But buttons don't implement the FormItem interface
- So on form Call AddButton
 - Pass button text and function to call when button pressed
 - Then set style on all buttons at the same time

```
form.AddButton("Previous", moveToPrev)  
form.AddButton("Next", moveToNext)  
form.AddButton("Quit", app.Stop)  
form.SetButtonBackgroundColor(tcell.ColorDarkGolden  
rod)
```

Responding to User Input



- Can respond to User input
 - Usually, keyboard – though mouse is possible
 - Handle your part of event and return event.

```
app.SetInputCapture(func(event *tcell.EventKey) *tcell.EventKey
{
    if event.Rune() == 'q' {
        app.Stop()
    }
    return event
})
```

Class discussion



- Go: what is it good for!?
- <https://www.infoworld.com/article/3198928/whats-the-go-language-really-good-for.html>
-
- We've looked at a variety of potential uses
 - What do you think?