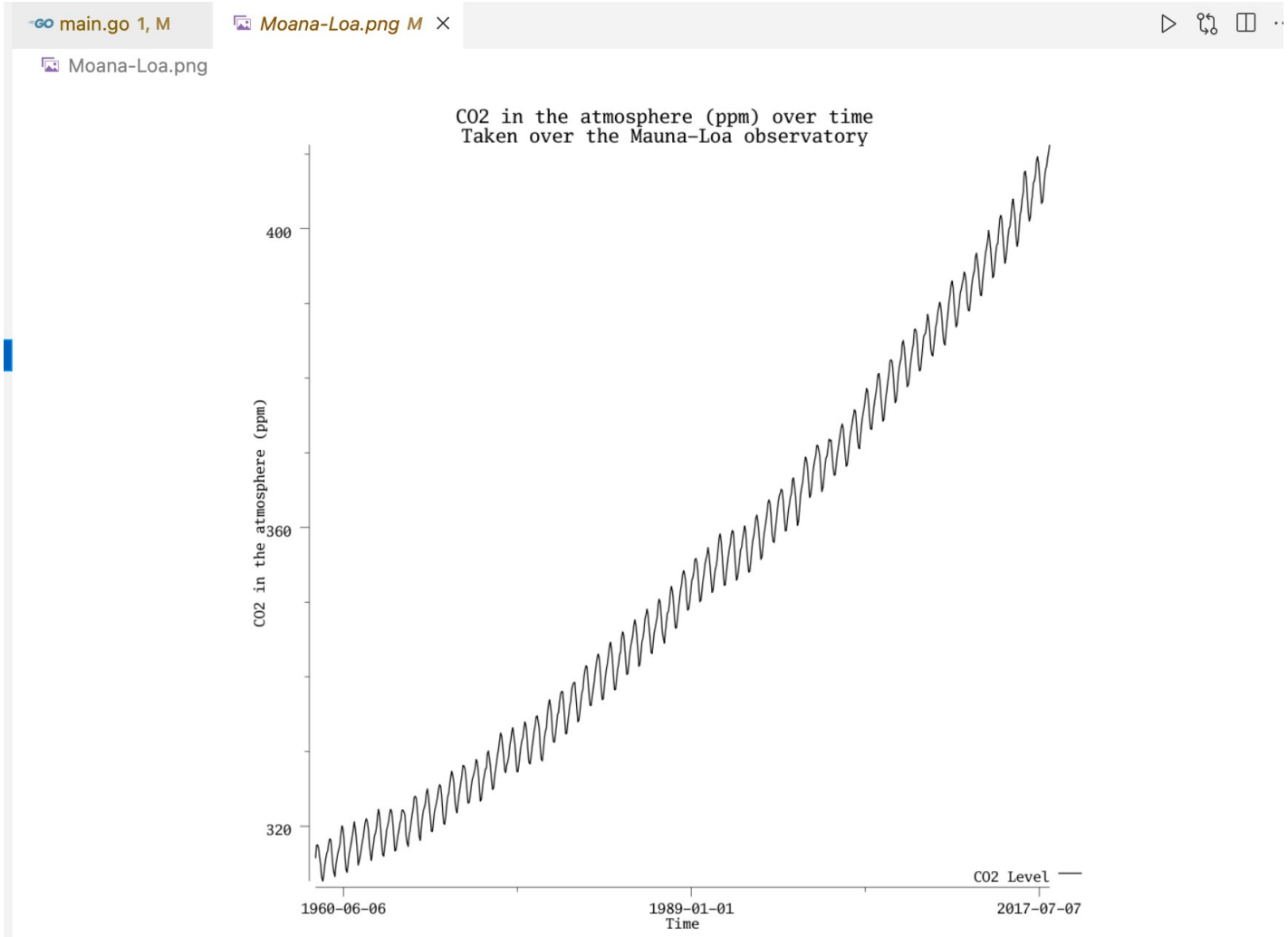
```
67   # CO2 expressed as a mole fraction in dry air, micromol/mol, abbreviated as ppm
68   #
69   #  (-99.99 missing data;  -1 no data for #daily means in month)
70   #
71   #            decimal     average   interpolated    trend    #days
72   #             date                              (season corr)
73   1958   3    1958.208      315.71      315.71      314.62     -1
74   1958   4    1958.292      317.45      317.45      315.29     -1
75   1958   5    1958.375      317.50      317.50      314.71     -1
76   1958   6    1958.458      -99.99      317.10      314.85     -1
77   1958   7    1958.542      315.86      315.86      314.98     -1
78   1958   8    1958.625      314.93      314.93      315.94     -1
79   1958   9    1958.708      313.20      313.20      315.91     -1
80   1958  10    1958.792      -99.99      312.66      315.61     -1
81   1958  11    1958.875      313.33      313.33      315.31     -1
82   1958  12    1958.958      314.67      314.67      315.61     -1
83   1959   1    1959.042      315.62      315.62      315.70     -1
84   1959   2    1959.125      316.38      316.38      315.88     -1
85   1959   3    1959.208      316.71      316.71      315.62     -1
```

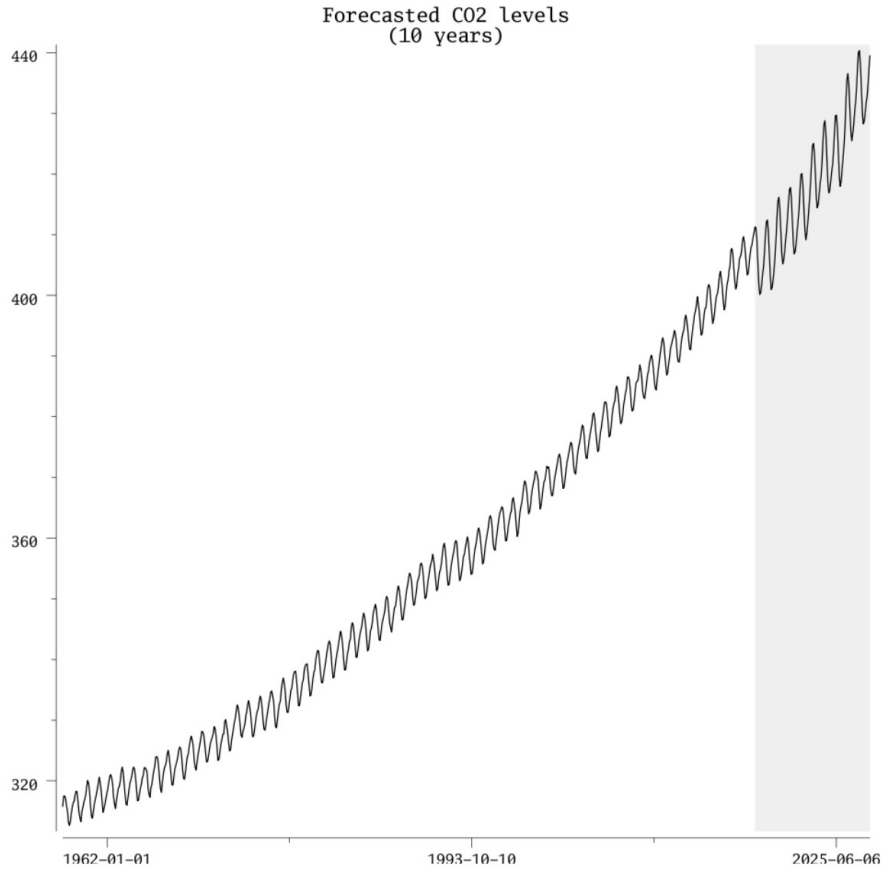
time series data values in file



plot time series data

main.go 1, M forecast.png M

forecast.png



time series data forecast

main.go 1, M forecast.png M

main.go > main

```
54  func main() {
55      dateStrings, co2s := parse(readFromFile)
56      fmt.Printf("%T", dateStrings)
57      fmt.Printf("%T", co2s)
58      dates := parseDates(dateStrings)
59      plt := newTSPlot(dates, co2s, "CO2 Level")
60      plt.X.Label.Text = "Time"
61      plt.Y.Label.Text = "CO2 in the atmosphere (ppm)"
62      plt.Title.Text = "CO2 in the atmosphere (ppm) over time\nTaken over the Mauna-Loa observatory"
63      dieIfErr(plt.Save(25*vg.Centimeter, 25*vg.Centimeter, "Moana-Loa.png"))
64
```

time series data plot go code

main.go 1, M | forecast.png M

main.go > main

```
80
81      fwd := 120
82      forecast := hw(decomposed, 12, fwd, 0.1, 0.05, 0.1)
83      datesplus := forecastTime(dates, fwd)
84      forecastPlot := newTSPlot(datesplus, forecast, "")
85      maxY := math.Inf(-1)
86      minY := math.Inf(1)
87      for i := range forecast {
88          if forecast[i] > maxY {
89              maxY = forecast[i]
90          }
91          if forecast[i] < minY {
92              minY = forecast[i]
93          }
94      }
95      //extend the range a little
96      minY--
97      maxY++
98      maxX := float64(datesplus[len(datesplus)-1].Unix())
99      minX := float64(datesplus[len(dates)-1].Unix())
100
101     shadePoly := plotter.XYs{
102         {X: minX, Y: minY},
103         {X: maxX, Y: minY},
104         {X: maxX, Y: maxY},
105         {X: minX, Y: maxY},
106     }
107     poly, err := plotter.NewPolygon(shadePoly)
108     dieIfErr(err)
109     poly.Color = color.RGBA{A: 16}
110     poly.LineStyle.Color = color.RGBA{}
111     forecastPlot.Add(poly)
112
113     writeToPng(forecastPlot, "Forecasted CO2 levels\n(10 years)", "forecast.png", 25, 25)
114 }
115
```

time series data forecast go code

A text to speech using go, docker, grpc, and kubernetes,

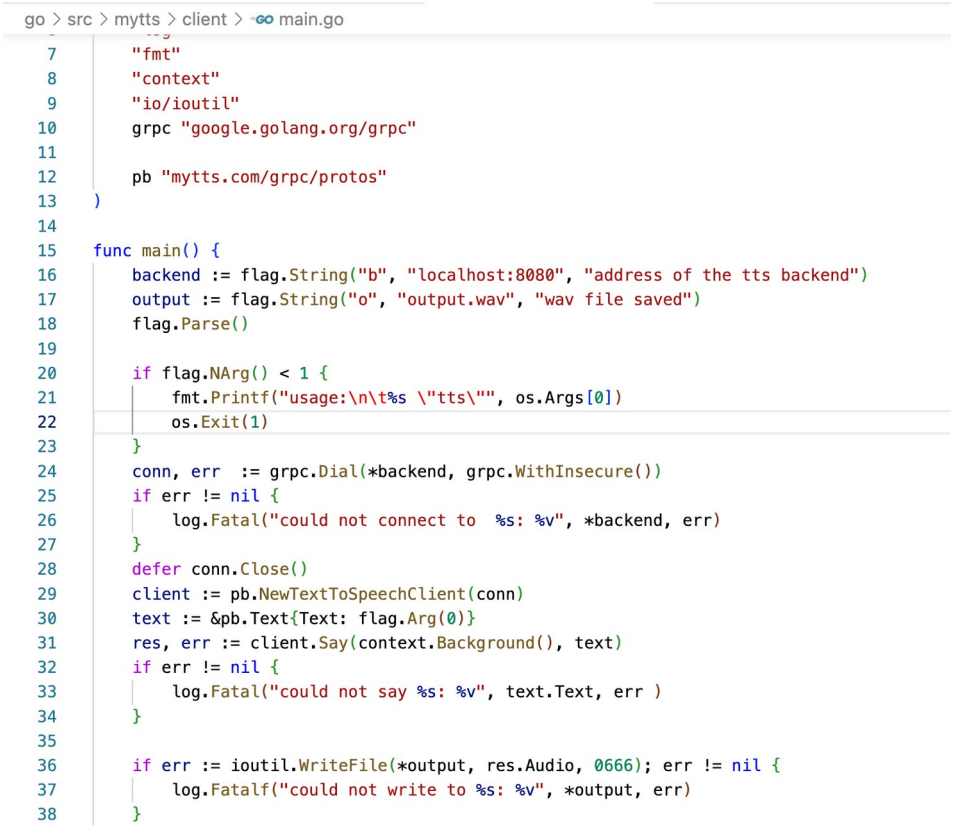
A text to speech grpc server – speech engine

go > src > mytts > protos > say.proto

```
1   syntax = "proto3";
2
3   option go_package = "myexample.com/grpc/protos";
4
5   package myexample;
6
7   service TextToSpeech {
8     rpc Say(Text) returns (Speech);
9   }
10
11  message Text {
12    string text = 1;
13  }
14
15  message Speech {
16    bytes Audio = 1;
17  }
18
```

A text to speech grpc server - proto

A text to speech grpc server - server

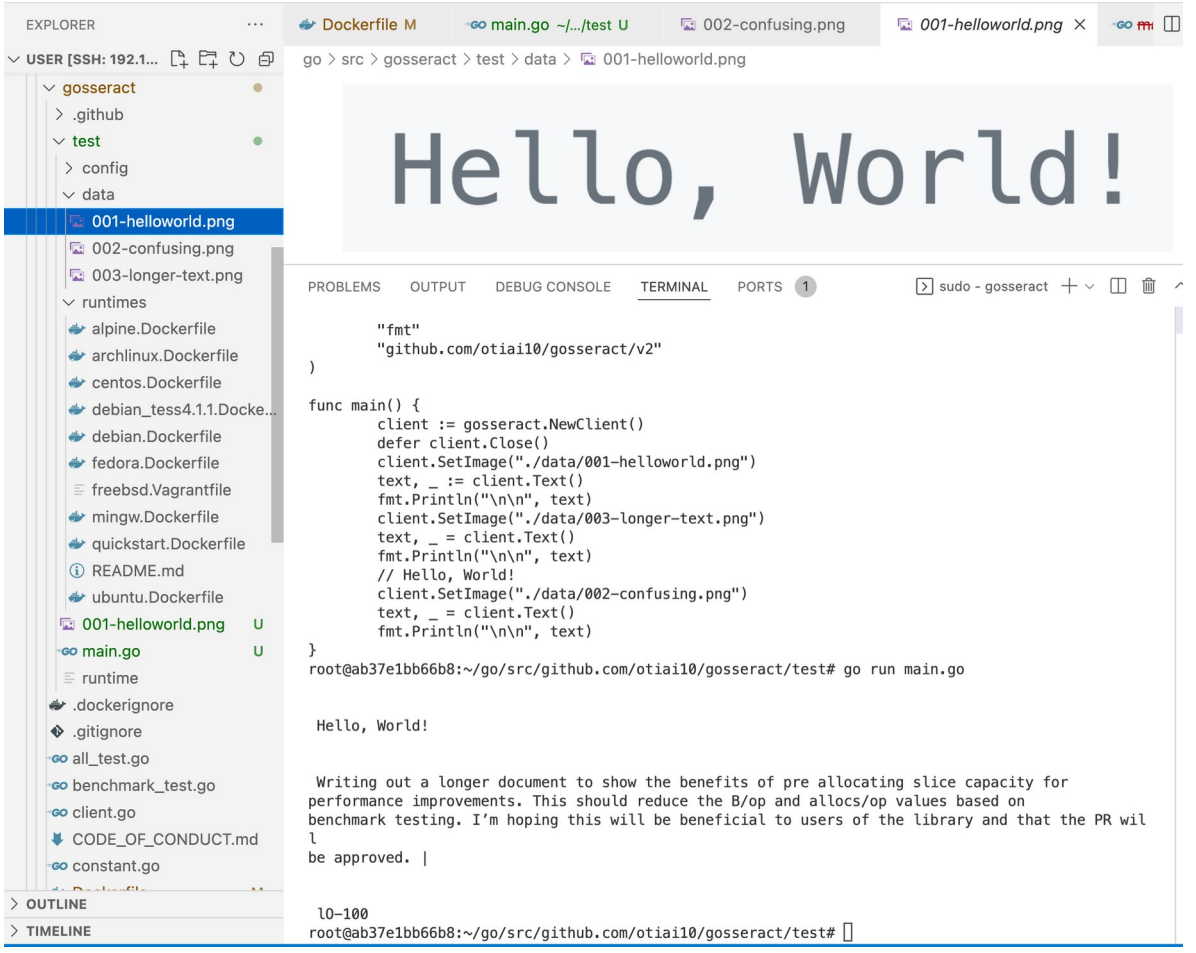


A text to speech grpc server - client

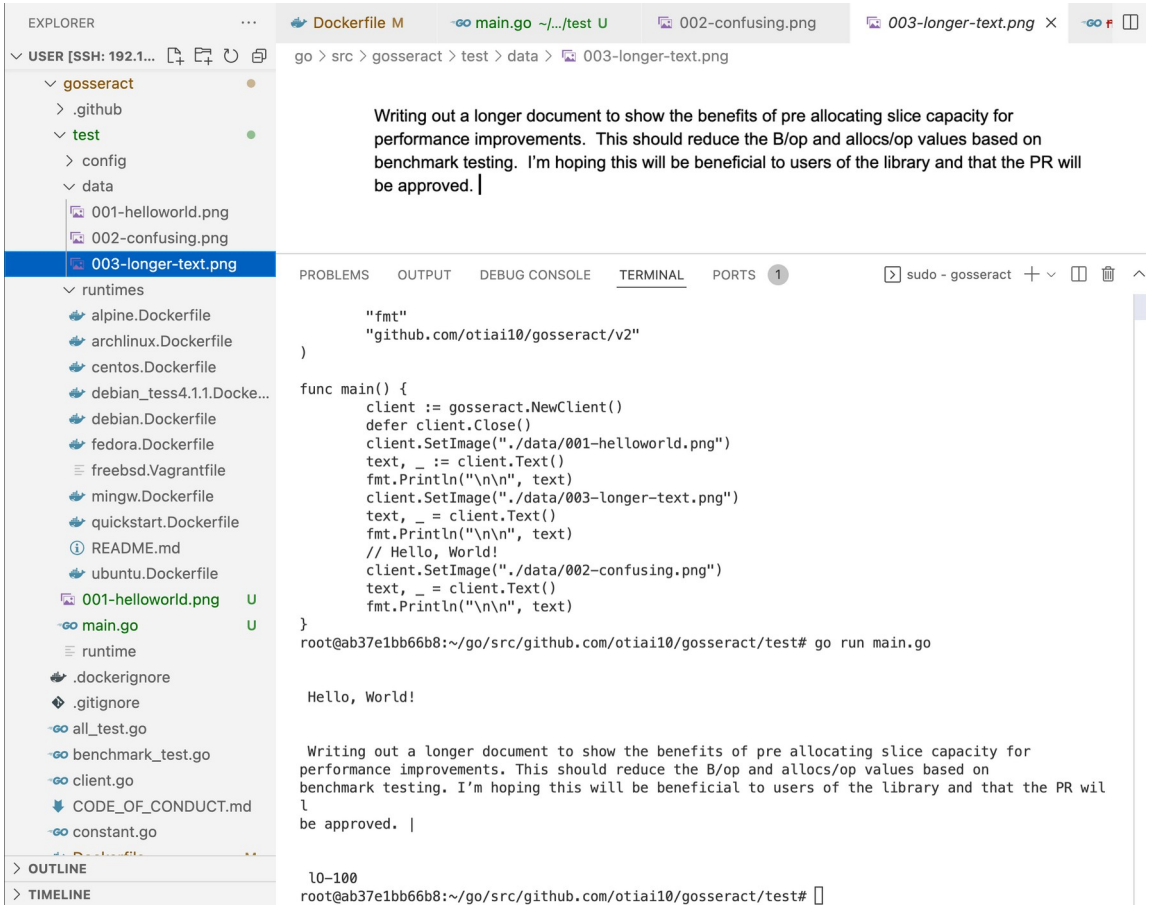
```
user@ml01:~/go/src/mytts/server$ sudo -E env "PATH=$PATH" make build
GOOS=linux go build -o app
docker build -t dev/tts .
Sending build context to Docker daemon  47.26MB
Step 1/5 : FROM ubuntu
 ---> 2dc39ba059dc
Step 2/5 : COPY app /app
 ---> 0c279ee50473
Step 3/5 : COPY flite /flite
 ---> 403ff0bc631f
Step 4/5 : RUN cp /flite /usr/local/bin
 ---> Running in 8f873ef9468c
Removing intermediate container 8f873ef9468c
 ---> a47b3c89e55d
Step 5/5 : ENTRYPOINT ["/app"]
 ---> Running in c5a80f50524e
Removing intermediate container c5a80f50524e
 ---> c20f0a1b6dd2
Successfully built c20f0a1b6dd2
Successfully tagged dev/tts:latest
#rm -f app
user@ml01:~/go/src/mytts/server$ sudo  docker run -p 8080:8080 dev/tts  --
 ls
hello
2022/09/04 19:35:41 server listening at %v [::]:8080
2022/09/04 19:35:41 &{{<nil> <nil> <nil> <nil> <nil> <nil> [] [] <nil> []
0 4194304 2147483647 <nil> {0 0 0 0 0} {0 false} 0 0 32768 32768 120000000
000 <nil> <nil> 0} {0 0} map[] map[] false false 0xc0000d09c0 map[myexampl
e.TextToSpeech:0xc0001668a0] <nil> 0xc0000b4440 0xc0000b4460 {0 {0 0}} {{}
 0 0} 0xc000166810 0xc0000c6360 []}
2022/09/04 19:35:45 received: text:"ai summit rocks"
user@ml01:~/go/src/mytts/server$
```

```
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
 rocks"
2022/09/04 20:32:
vailable desc = c
l tcp 127.0.0.1:8
exit status 1
user@ml01:~/go/sr
 rocks"
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
 rocks"
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
```

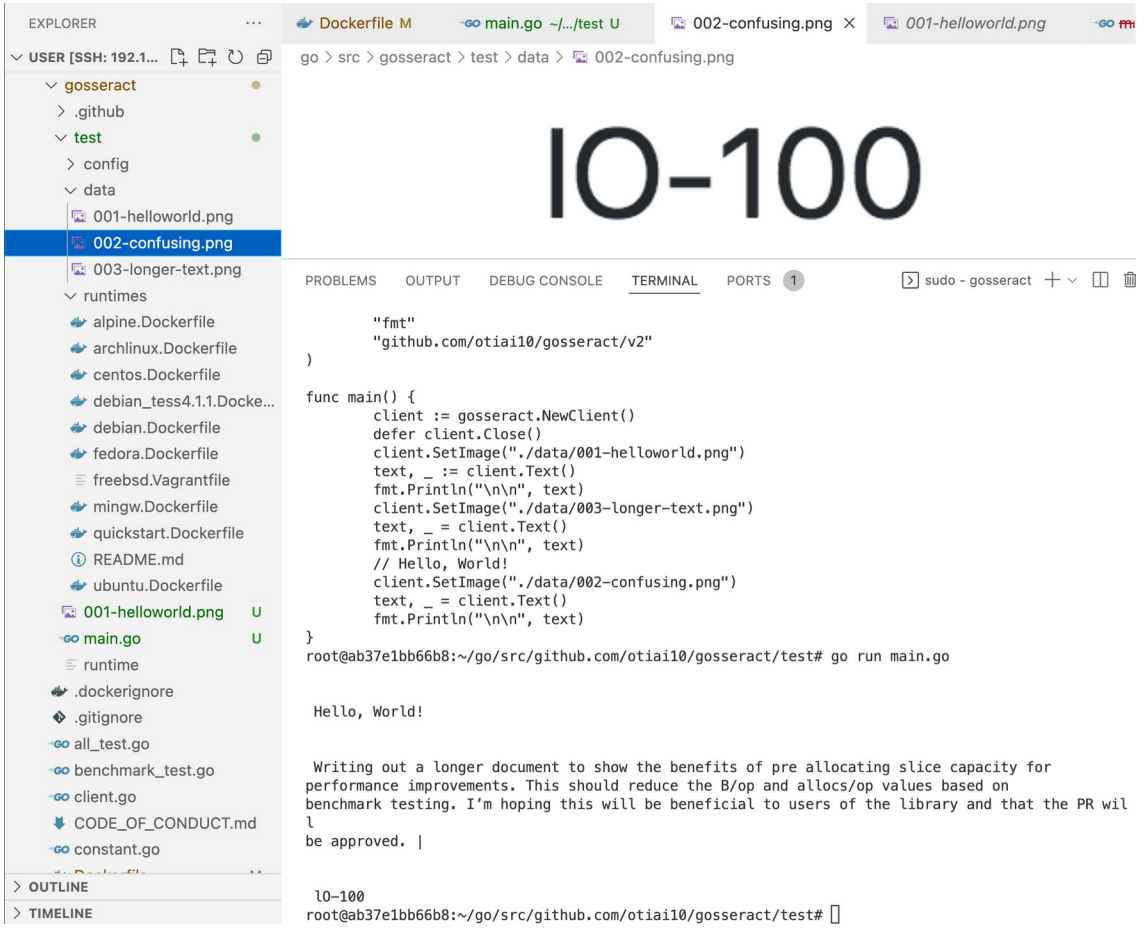
A text to speech grpc server - test results



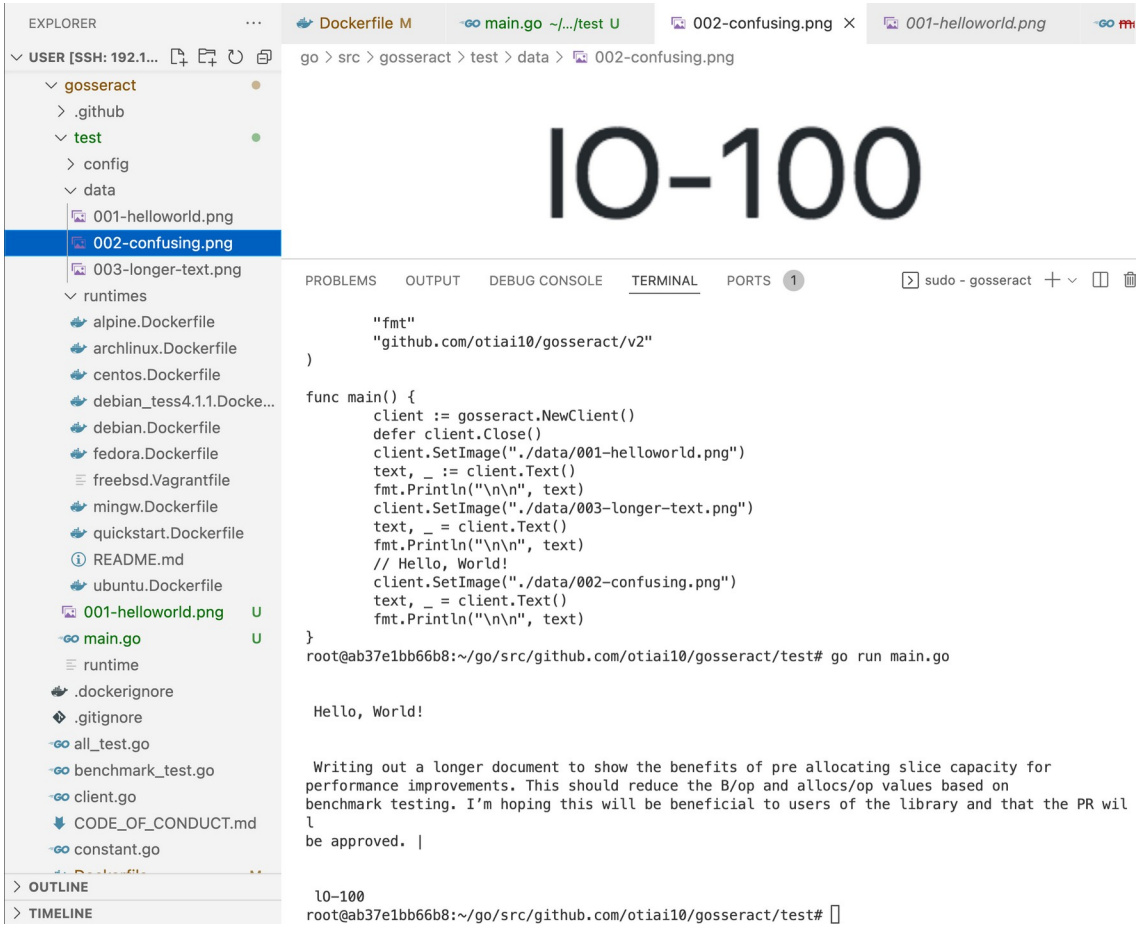
An ocr app using tesseract, go / c-binding, docker – hello world



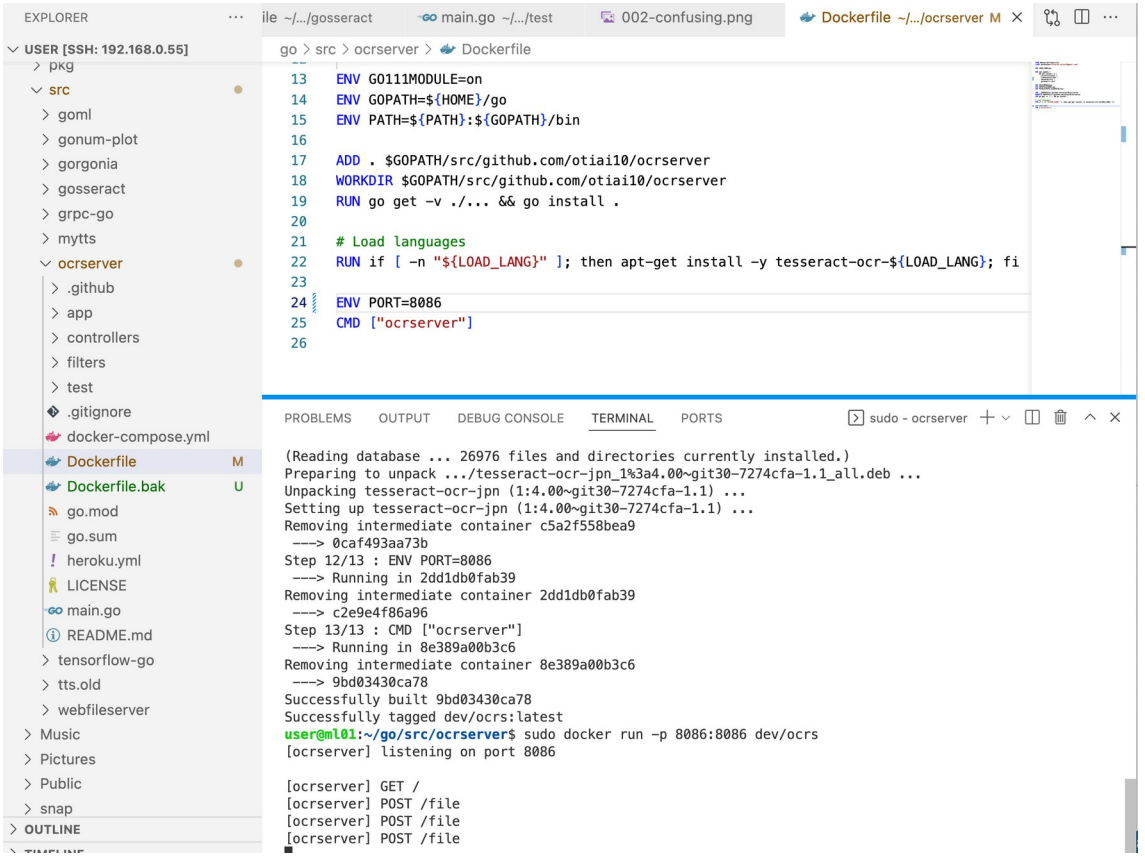
An ocr app using tesseract, go / c-binding, docker – long text



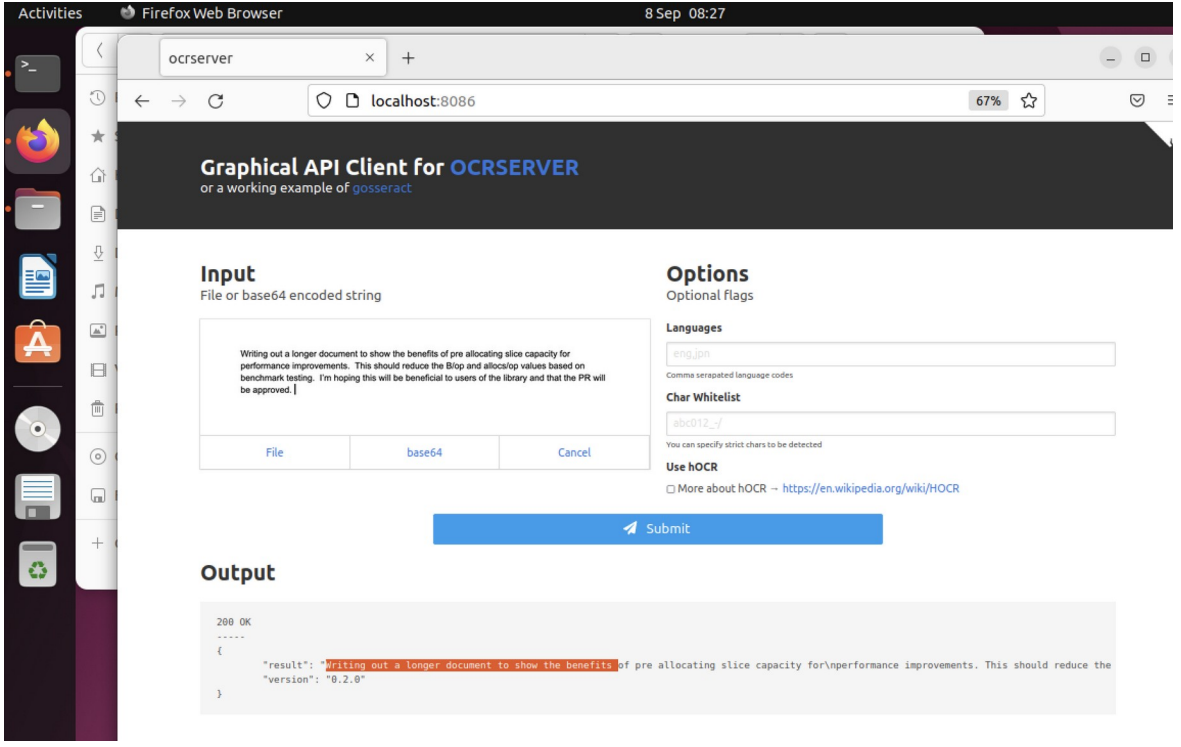
An ocr app using tesseract, go / c-binding, docker – confusing text



An ocr app using tesseract, go / c-binding, docker – confusing text



An ocr app using tesseract, go / c-binding, docker – build / run docker server



An ocr app using tesseract, go / c-binding, docker – docker ocr server gui and results

```
user@ml01:~/go/src/mytts/server$ sudo -E env "PATH=$PATH" make build
GOOS=linux go build -o app
docker build -t dev/tts .
Sending build context to Docker daemon  47.26MB
Step 1/5 : FROM ubuntu
 ---> 2dc39ba059dc
Step 2/5 : COPY app /app
 ---> 0c279ee50473
Step 3/5 : COPY flite /flite
 ---> 403ff0bc631f
Step 4/5 : RUN cp /flite /usr/local/bin
 ---> Running in 8f873ef9468c
Removing intermediate container 8f873ef9468c
 ---> a47b3c89e55d
Step 5/5 : ENTRYPOINT ["/app"]
 ---> Running in c5a80f50524e
Removing intermediate container c5a80f50524e
 ---> c20f0a1b6dd2
Successfully built c20f0a1b6dd2
Successfully tagged dev/tts:latest
#rm -f app
user@ml01:~/go/src/mytts/server$ sudo  docker run -p 8080:8080 dev/tts  --
 ls
hello
2022/09/04 19:35:41 server listening at %v [::]:8080
2022/09/04 19:35:41 &{{<nil> <nil> <nil> <nil> <nil> <nil> [] [] <nil> []
0 4194304 2147483647 <nil> {0 0 0 0 0} {0 false} 0 0 32768 32768 120000000
000 <nil> <nil> 0} {0 0} map[] map[] false false 0xc0000d09c0 map[myexampl
e.TextToSpeech:0xc0001668a0] <nil> 0xc0000b4440 0xc0000b4460 {0 {0 0}} {{}
 0 0} 0xc000166810 0xc0000c6360 []}
2022/09/04 19:35:45 received: text:"ai summit rocks"
user@ml01:~/go/src/mytts/server$
```

```
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
 rocks"
2022/09/04 20:32:
vailable desc = c
l tcp 127.0.0.1:8
exit status 1
user@ml01:~/go/sr
 rocks"
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
 rocks"
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
```

A text to speech grpc server - test results

```
user@ml01:~/go/src/mytts/server$ sudo -E env "PATH=$PATH" make build
GOOS=linux go build -o app
docker build -t dev/tts .
Sending build context to Docker daemon  47.26MB
Step 1/5 : FROM ubuntu
 ---> 2dc39ba059dc
Step 2/5 : COPY app /app
 ---> 0c279ee50473
Step 3/5 : COPY flite /flite
 ---> 403ff0bc631f
Step 4/5 : RUN cp /flite /usr/local/bin
 ---> Running in 8f873ef9468c
Removing intermediate container 8f873ef9468c
 ---> a47b3c89e55d
Step 5/5 : ENTRYPOINT ["/app"]
 ---> Running in c5a80f50524e
Removing intermediate container c5a80f50524e
 ---> c20f0a1b6dd2
Successfully built c20f0a1b6dd2
Successfully tagged dev/tts:latest
#rm -f app
user@ml01:~/go/src/mytts/server$ sudo  docker run -p 8080:8080 dev/tts  --
 ls
hello
2022/09/04 19:35:41 server listening at %v [::]:8080
2022/09/04 19:35:41 &{{<nil> <nil> <nil> <nil> <nil> <nil> [] [] <nil> []
0 4194304 2147483647 <nil> {0 0 0 0 0} {0 false} 0 0 32768 32768 120000000
000 <nil> <nil> 0} {0 0} map[] map[] false false 0xc0000d09c0 map[myexampl
e.TextToSpeech:0xc0001668a0] <nil> 0xc0000b4440 0xc0000b4460 {0 {0 0}} {{}
 0 0} 0xc000166810 0xc0000c6360 []}
2022/09/04 19:35:45 received: text:"ai summit rocks"
user@ml01:~/go/src/mytts/server$
```

```
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
 rocks"
2022/09/04 20:32:
vailable desc = c
l tcp 127.0.0.1:8
exit status 1
user@ml01:~/go/sr
 rocks"
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
 rocks"
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
Playing WAVE 'out
user@ml01:~/go/sr
```

A text to speech grpc server - test results