Go Documentation Server

Package routing

```
import
"TanJunJie_GoInAction2Submission/TanJunJie_GoInAction2Submission/routing"
```

Overview

Package routing routes matching incoming requests to their respective handlers.

Index

Package files

admin.go bookNewAppt.go chooseDoctor.go choosePatient.go doc.go editAppointment.go index.go login.go logout.go route.go searchAppointment.go searchAvailableDoctors.go searchDoctorAvailability.go signup.go user.go viewDoctorAppointments.go viewPatientAppointmentHistory.go

Variables

Server uses these variables to store data and perform logging actions.

```
var (
    doctors     system.Doctors
    patients    system.Patients
    mapUsers    = map[string]userInfo{}
    mapSessions = map[string]string{}
    tpl         *template.Template
    r           result
    file        *os.File

    InfoLogger *log.Logger
    WarnLogger *log.Logger
    ErrLogger  *log.Logger
)
```

```
var wg sync.WaitGroup
```

```
var wgEdit sync.WaitGroup
```

func **Route**

```
func Route(port int)
```

Route starts a web server at http://127.0.0.1/(port number) or localhost:port with a self-signed certificate. Logs generated by route are placed in the "logs" directory.

func **admin**

```
func admin(res http.ResponseWriter, req *http.Request)
```

admin is a handler that renders the admin page.

func **bookNewAppointment**

```
func bookNewAppointment(res http.ResponseWriter, req *http.Request)
```

bookNewAppointment is a handler that renders the booking page.

func **chooseDoctor**

```
func chooseDoctor(res http.ResponseWriter, req *http.Request)
```

chooseDoctor is a handler that renders the doctor selection page.

func **choosePatient**

```
func choosePatient(res http.ResponseWriter, req *http.Request)
```

choosePatient is a handler that renders the patient selection page.

func **editAppointment**

```
func editAppointment(res http.ResponseWriter, req *http.Request)
```

editAppointment is a handler that renders the appointment editing page.

func **index**

```
func index(res http.ResponseWriter, req *http.Request)
```

index redirects users to the login page. The default landing page for the application is the login page. If the user is logged in, the user will be redirected to either the admin page or the home page.

func **init**

```
func init()
```

init will only be called once.

func **isAdmin**

```
func isAdmin(res http.ResponseWriter, req *http.Request) bool
```

isAdmin checks if the logged-in user is an admin.

func **isLoggedIn**

```
func isLoggedIn(req *http.Request) bool
```

isLoggedIn checks if a user is logged in based on the username retrieved from loginCookie.

func **login**

```
func login(res http.ResponseWriter, req *http.Request)
```

login is a handler that renders the login page. If the user is logged in, the user will be redirected to either the admin page or the home page.

func **logout**

```
func logout(res http.ResponseWriter, req *http.Request)
```

logout logs a user out of the application and redirects them to the index page.

func **searchAppointment**

```
func searchAppointment(res http.ResponseWriter, req *http.Request)
```

searchAppointment is a handler that loads the appointment searching page. Users are redirected to login if they are not logged in. Users are redirected to the appointment editing page if they key in a valid patient name into the form.

func **searchAvailableDoctors**

```
func searchAvailableDoctors(res http.ResponseWriter, req *http.Request)
```

searchAvailableDoctors is a handler that loads the doctor's availability search page.

func **searchDoctorAvailability**

```
func searchDoctorAvailability(res http.ResponseWriter, req *http.Request)
```

searchDoctorAvailability is a handler that displays the doctor's availability page.

func **signup**

```
func signup(res http.ResponseWriter, req *http.Request)
```

signup is a handler that renders the signup page. signup redirects a user to the index page if the user is already logged in.

func **user**

```
func user(res http.ResponseWriter, req *http.Request)
```

user is a handler that renders a non-admin user's home page.

func **viewDoctorAppointments**

```
func viewDoctorAppointments(res http.ResponseWriter, req *http.Request)
```

viewDoctorAppointments is a handler that renders a doctor's schedule on the page.

func **viewPatientAppointmentHistory**

```
func viewPatientAppointmentHistory(res http.ResponseWriter, req *http.Request)
```

viewPatientAppointmentHistory is a handler that renders a patient's appointment history on the page.

type **result**

result contains fields used to set values in the respective .gohtml files.

```
type result struct {
    User                userInfo
    DoctorsAvailability template.HTML
    DoctorAvailability  template.HTML
    BookingStatus       bool
    PatientsName        template.HTML
    PatientHistory      template.HTML
    DoctorsName         template.HTML
    DoctorAppointments  template.HTML
    DoctorName          template.HTML
    PatientName         template.HTML
}
```

type **userInfo**

userInfo contains information regarding a user.

```
type userInfo struct {
    Username  string
    Password  []byte
    FirstName string
    LastName  string
}
```

func getUser

```
func getUser(res http.ResponseWriter, req *http.Request) userInfo
```

getUser retrieves the user information if the user is logged in.

Build version go1.16.3.