**Requirements**

A timesheet approval system that:

1. Allows for the review of Entered, Approved, and Rejected timesheets
2. All timesheets are stored within a MongoDB database
3. Allows Entered timesheets to be approved or rejected.
4. Allows Rejected and Approved to be reverted back to an Entered state
5. Any change of state (from Entered to Approved/Rejected and Approved/Rejected back to Entered) caused an email to be generated

**Technical Stack**

1. Latest Ionic Framework version
2. MongoDB (v 6.0) and MongoDB Compass (v1.34.1)
3. Flask, pymongo and flask\_cors

**Zip Files**

1. TimesheetIonic
   1. src
      1. App
         1. Pages
            1. about

about.page.html

about.module.ts

about.module.scss

about.page.spec.ts

about.page.ts

about.routing.module.ts

* + - * 1. Approved

approved-routing.module.ts

approved.module.ts

approved.page.html

approved.page.scss

approved.page.spec.ts

approved.page.spec.ts

* + - * 1. Home

home-routing.module.ts

home.module.ts

home.page.html

home.page.scss

home.page.spec.ts

home.page.ts

* + - * 1. Rejected

rejected-routing.module.ts

rejected.module.ts

rejected.page.html

rejected.page.scss

rejected.page.spec.ts

rejected.page.ts

* + - * 1. tabs

tabs-routing.module.ts

tabs.module.ts

tabs.page.html

tabs.page.scss

tabs.page.spec.ts

tabs.page.ts

* + 1. Providers
       1. Timesheet.service.service
          1. timesheet-service-service.ts
          2. timesheet-service-service.spec
    2. App-component files (app-component.css, app-component.spec.ts, app-component.ts, app-module.ts, app-routing-module.ts)

1. Server
   1. runflask.cmd (sets Flask environment variable and then runs flask)
   2. timeSheetPrint.py
   3. timeSheetResetAll.py
   4. timeSheetRest.py
   5. timeSheetTest.py