

INFS809 SDM Assignment 1B Evidences

Contents

1 VS Code, Git repo on own machine	1
2 GitHub repository	2
3 Gitflow branching configuration	3
4 pull request function of GitHub	3
5 TravisCI	4
6 Node.js, React, express.js and MongoDB	5
7 Mocha and Chai	6
8 Deployment to Heroku	7
9 Cucumber	8
10 Docker	9

1 VS Code, Git repo on own machine

screenshot of VS code and local git repo:

The screenshot shows the Visual Studio Code interface. The code editor has `index.js` open, displaying the following code:

```

const express = require('express')
const app = express()
const port = 3000

app.get('/', (req, res) => res.send('Hello World!'))

app.listen(port, () => console.log(`Example app listening on port ${port}!`))

```

The terminal window shows the following output:

```

nothing to commit, working tree clean
stone sdm2019-ass1b% git status
On branch develop
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   doc/portfolio.md

no changes added to commit (use "git add" and/or "git commit -a")
stone sdm2019-ass1b% git branch
* develop
  feature/f-error
  feature/f-pr
  master
stone sdm2019-ass1b%

```

2 GitHub repository

link: <https://github.com/nanoix9/sdm2019-ass1b>

screenshot:

The screenshot shows a GitHub repository page for `nanoix9/sdm2019-ass1b`. The repository summary section includes:

- 22 commits
- 10 branches
- 3 releases
- 1 contributor
- MIT license

A pull request titled "Merge pull request #3 from nanoix9/develop" is highlighted. The commit history shows the following changes:

File	Commit Message	Date
<code>doc</code>	add index.html	4 days ago
<code>src</code>	setup node.js project and add entry file	4 days ago
<code>test</code>	add test entry	4 days ago
<code>.gitignore</code>	Initial commit	4 days ago
<code>.travis.yml</code>	add travis file	4 days ago
<code>LICENSE</code>	Initial commit	4 days ago
<code>README.md</code>	edit readme	4 days ago
<code>index.html</code>	test pull request	4 days ago
<code>package-lock.json</code>	setup node.js project and add entry file	4 days ago

3 Gitflow branching configuration

Git flow makes use of several branch naming conventions: master, develop, feature, hotfix, release.

link: <https://github.com/nanoix9/sdm2019-ass1b/branches/all>

screenshot:

Branch	Last Updated	Commits	Status
master	4 days ago by nanoix9	✓	Default
develop	4 days ago by nanoix9	✓	5 0 #3 Merged
feature/nodejs	4 days ago by nanoix9	✓	7 0 New pull request
feature/f-error	4 days ago by nanoix9	✓	8 1 #2 Closed
feature/f-pr	4 days ago by nanoix9	✓	7 0 #1 Merged
feature/f3	4 days ago by nanoix9	✓	14 0 New pull request
feature/f2	4 days ago by nanoix9	✓	17 0 New pull request
release/0.0.1	4 days ago by nanoix9	✓	17 0 New pull request
feature/readme	4 days ago by nanoix9	✓	20 0 New pull request
feature/helloworld	4 days ago by nanoix9	✓	20 0 New pull request

4 pull request function of GitHub

link:

- <https://github.com/nanoix9/sdm2019-ass1b/pulls?q=is%3Apr+is%3Aclosed>
- <https://github.com/nanoix9/sdm2019-ass1b/pull/1>

screenshots:

succeeded:

A screenshot of a GitHub pull request page for a repository named "sdm2019-ass1b". The pull request is titled "test pull request #1" and is in a "Merged" state. It was merged by nanoix9 from the "feature/f-pr" branch into the "master" branch 4 days ago. The commit message is "test pull request". The pull request has 1 commit, 0 checks, and 1 file changed. The right sidebar shows various project details like Reviewers, Assignees, Labels, Projects, Milestone, and Notifications.

rejected:

A screenshot of a GitHub pull request page for a repository named "sdm2019-ass1b". The pull request is titled "error commit #2" and is in a "Closed" state. It was closed by nanoix9 4 days ago. The commit message is "error commit". The pull request has 1 commit, 0 checks, and 1 file changed. The right sidebar shows various project details like Reviewers, Assignees, Labels, Projects, Milestone, and Notifications. A note at the bottom states: "Closed with unmerged commits. This pull request is closed, but the feature/f-error branch has unmerged commits."

5 TravisCI

To enable TravisCI, developer should put a `.travis.yml` in github repo and connect travis-ci.com with github account.

link: <https://github.com/nanoix9/sdm2019-ass1b/blob/master/.travis.yml>

screenshot of one TravisCI job details:

The screenshot shows a GitHub pull request page for the repository 'nanoix9 / sdm2019-ass1b'. The pull request has been merged and is now a branch named 'develop'. A successful Travis CI build is shown, indicating the merge commit was run in half a minute. The build details show it ran 7 minutes ago and was triggered by a merge from the 'develop' branch. The build status is green, labeled 'Build Passed'. A note states: 'The build passed, just like the previous build.' Below the build details, a 'DETAILS' section notes: 'This is a pull request build. It is running a build against the merge commit, after merging #7 Develop. Any changes that have been made to the master branch before the build ran are also included.'

6 Node.js, React, express.js and MongoDB

create an application that show a simple table. To start the application, run `npm start`

- node.js: <https://github.com/nanoix9/sdm2019-ass1b/blob/master/package.json>
- react: <https://github.com/nanoix9/sdm2019-ass1b/blob/master/src/app/app.jsx>
- express: <https://github.com/nanoix9/sdm2019-ass1b/blob/master/src/index.js>
- mongodb: <https://github.com/nanoix9/sdm2019-ass1b/blob/master/src/server/db.js>

screenshot of source code:

```

src > server > JS server.js > ...
1 const express = require('express')
2 const path = require('path');
3 const db = require('./db.js');
4
5 const app = express()
6
7 const res_path = path.join(path.dirname(path.dirname(__dirname)),
8
9 app.set('views', res_path);
10 app.engine('html', require('ejs').renderFile);
11 app.use(express.static(res_path));
12
13 app.get('/hello', (req, res) => res.send('Hello World!'))
14
15 app.get('/', (req, res) => {
  res.send('Hello World!')
})

src > server > JS db.js > MongoClient
1 const config = require('.../config/config.js')[process.env.NODE_ENV]
2
3 const MongoClient = require('mongodb').MongoClient
4
5 const client = new MongoClient(config.MONGO_URI);
6
7 const find_all = (collection_name) => {
8   return new Promise((resolve, reject) => {
9     client.connect((err) => {
10       if (err) {
11         console.debug('mongodb error: ', err);
12         reject(err);
13       }
14     })
15   })
}

src > app > JS app.jsx > App > constructor
1 import React from 'react';
2
3 class App extends React.Component {
4
5   constructor(props) {
6     super(props)
7     this.state = {
8       users: []
9     }
10
11   componentWillMount() {
12     fetch('/users')
13       .then((res) => {
14         console.debug(res)
15       })
16   }
17
18   render() {
19     return (
20       <div>
21         <h1>Welcome to SDM Assignment 1B</h1>
22         <table border="1">
23           <thead>
24             <tr>
25               <th>ID</th>
26               <th>Name</th>
27             </tr>
28           </thead>
29           <tbody>
30             {this.state.users.map((user) => {
31               return <tr key={user._id}>
32                 <td>{user._id}</td>
33                 <td>{user.name}</td>
34               </tr>
35             })}
36           </tbody>
37         </table>
38       </div>
39     )
40   }
41 }

index.html > views > index.html > html > style
1 <html>
2   <style>
3     table {
4       border-collapse: collapse;
5     }
6
7     table,
8     th,
9     td {
10       border: 1px solid black;
11     }
12   </style>
13   <header><title>React App for SDM Assignment 1B</title></header>
14   <body>
15     <div id="root"></div>

```

web page:

Docs from MongoDB

ID	Name
1	foo
2	bar

```

d configuration is unrestricted.
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten]
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] ** WARNING: t.
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] **
connect to this server.
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] **
<address> to specify which IP
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] **
ses from, or with --bind_ip_all to
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] **
behavior is desired, start the
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] **
to disable this warning.
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten]
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten]
2019-08-19T14:53:52.567+1200 I CONTROL [initandlisten] ** WARNING: s
files is 256, should be at least 1000
---
Enable MongoDB's free cloud-based monitoring service, which will then
metrics about your deployment (disk utilization, CPU, operation stati
The monitoring data will be available on a MongoDB website with a un
and anyone you share the URL with. MongoDB may use this information to
improvements and to suggest MongoDB products and deployment options.
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---

> db.user.find()
[ { "_id" : 2, "name" : "bar" },
  { "_id" : 1, "name" : "foo" } ]
>

```

7 Mocha and Chai

Mocha is a unit test framework and Chai is an assertion library

command:

```
npx mocha
```

link: <https://github.com/nanoix9/sdm2019-ass1b/blob/master/test/test.js>

screenshot:

The screenshot shows the Visual Studio Code interface. The left sidebar displays the project structure with files like README.md, index.html, index.js, portfolio.md, package.json, package-lock.json, test.js, .travis.yml, img, node_modules, src, and test. The right pane shows the content of package.json, which includes dependencies for express (~4.17.1), chai (^4.2.0), and mocha (^6.2.0). Below the code editor is the terminal window, which shows the command 'npx mocha' being run. The output in the terminal shows the test results: 1 passing (14ms). The bottom status bar indicates the file is 1: zsh.

```
package.json — sdm2019-ass1b
17 "author": "Stone Fang",
18 "license": "MIT",
19 "bugs": {
20   "url": "https://github.com/nanoix9/sdm2019-ass1b/issues"
21 },
22 "homepage": "https://github.com/nanoix9/sdm2019-ass1b#readme",
23 "dependencies": {
24   "express": "^4.17.1"
25 },
26 "devDependencies": {
27   "chai": "^4.2.0",
28   "mocha": "^6.2.0"
29 }
30 }
```

```
stone sdm2019-ass1b% npm test
> sdm2019-ass1b@1.0.0 test /Users/stone/aut/sdm2019-ass1b
> ./node_modules/mocha/bin/mocha
example
use assert
  ✓ should be numbers
  1 passing (14ms)
```

8 Deployment to Heroku

Heroku is a cloud platform to host web applications

command:

```
heroku create
git push heroku develop:master
```

link: <https://lit-beyond-33040.herokuapp.com>

screenshot:

The screenshot shows a macOS desktop environment. In the foreground, a Terminal window is open with the title "Terminal - zsh - %1". It displays a command-line session where a user is pushing code to a Heroku remote. The output includes messages about object enumeration, compression, and the creation of a runtime environment. It also shows the installation of Node.js and npm, and the auditing of dependencies. The background shows a code editor with files like "dex.jsx", "webpack.config.js", "package.json", and "index.js" visible. The status bar at the bottom of the screen provides information about the file "portfolio.md", including line and column numbers (Ln 47, Col 41), spaces used (Spaces: 4), and encoding (UTF-8). It also shows icons for Go Live and Prettier.

9 Cucumber

Cucumber is a tool to write test in natural language, which usually contains three kind of files: feature file, world file, and steps file.

command:

```
npx cucumber-js
```

code link: <https://github.com/nanoix9/sdm2019-ass1b/tree/master/features>

screenshot:

```

Feature: Hello world API
  Scenario: hello world
    When I send get request to "/hello"
    Then I should get "Hello World!"

const { Given, When, Then } = require("cucumber");
const { expect, assert } = require("chai");
When("I send get request to {string}", function(path) {
  this.sendRequest(path);
});
Then("I should get {string}", async function(text) {
  const res_text = await this.res_json;
  expect(res_text).to.eql(text);
});

Feature: Support
  Scenario: Support
    When I send get request to "/support"
    Then I should get "Support"

const { Given, When, Then } = require("cucumber");
When("I send get request to {string}", function(path) {
  this.sendRequest(path);
});
Then("I should get {string}", function(text) {
  expect(this.res_json).to.eql(text);
});
  
```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL

```

/Users/stone/.zshrc:source:3: no such file or directory: .bashrc
stone sdm2019-ass1b% npm run cuteset
> sdm2019-ass1b@1.0.0 cuteset /Users/stone/aut/sdm2019-ass1b
> cucumber-js
...
1 scenario (1 passed)
2 steps (2 passed)
0m00.024s
stone sdm2019-ass1b% 
```

10 Docker

Add a dockerfile first: <https://github.com/nanoix9/sdm2019-ass1b/tree/master/Dockerfile>

build docker image:

```
docker build -t stone/sdm-ass1b .
```

run docker container:

```
docker run -p 3000:3000 --rm stone/sdm-ass1b
```

screenshot:

The screenshot shows a terminal window with the following content:

```
Terminal - zsh — %1
mongod
example
use assert
✓ should be numbers

1 passing (15ms)

Removing intermediate container 5faf684a8083
--> 6f20bc39a7d9
Step 8/9 : EXPOSE 3000
--> Running in 78b1658e813d
Removing intermediate container 78b1658e813d
--> 1a0c372fde45
Step 9/9 : CMD [ "npm", "run", "startbackend" ]
--> Running in 5bde67fa8755
Removing intermediate container 5bde67fa8755
--> f23cef6a873d
Successfully built f23cef6a873d
Successfully tagged stone/sdm-ass1b:latest
stone sdm2019-ass1b% docker run --rm stone/sdm-ass1b
> sdm2019-ass1b@1.0.0 startbackend /usr/src/app
> node src/index.js

using MongoDB at mongodb://localhost/test database undefined
Example app listening on port 3000!
^C
stone sdm2019-ass1b% docker run -p 3000:3000 --rm stone/sdm-ass1b
> sdm2019-ass1b@1.0.0 startbackend /usr/src/app
> node src/index.js

using MongoDB at mongodb://localhost/test database undefined
Example app listening on port 3000!
```

The terminal window also displays the file structure in the left sidebar, including files like package.json, .babelrc, config.js, db.js, Dockerfile, and Dockerignore.