

ZHANCHAO (EDMOND) WEN

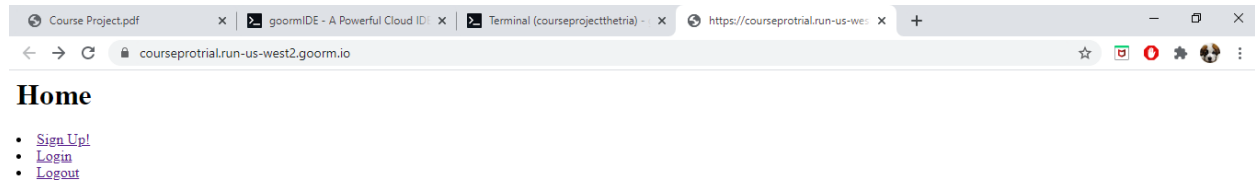
INFO 1213 (R95)

Description of the Business

The business I have decided to do for my project is a restaurant business. It would be providing food to people who want to eat takeout or people that are just lazy in general to cook for themselves. The potential customers would be anyone that is in a rush and has no time to be able to cook food for themselves. Other than that, just foodies which are people that want to try different types of cuisines. As well as a restaurant that has been in business for three generations bring fresh Chinese food to your table. People would use this to get food without a hassle. As well as a restaurant that is open to selling dim sum which people love to enjoy.

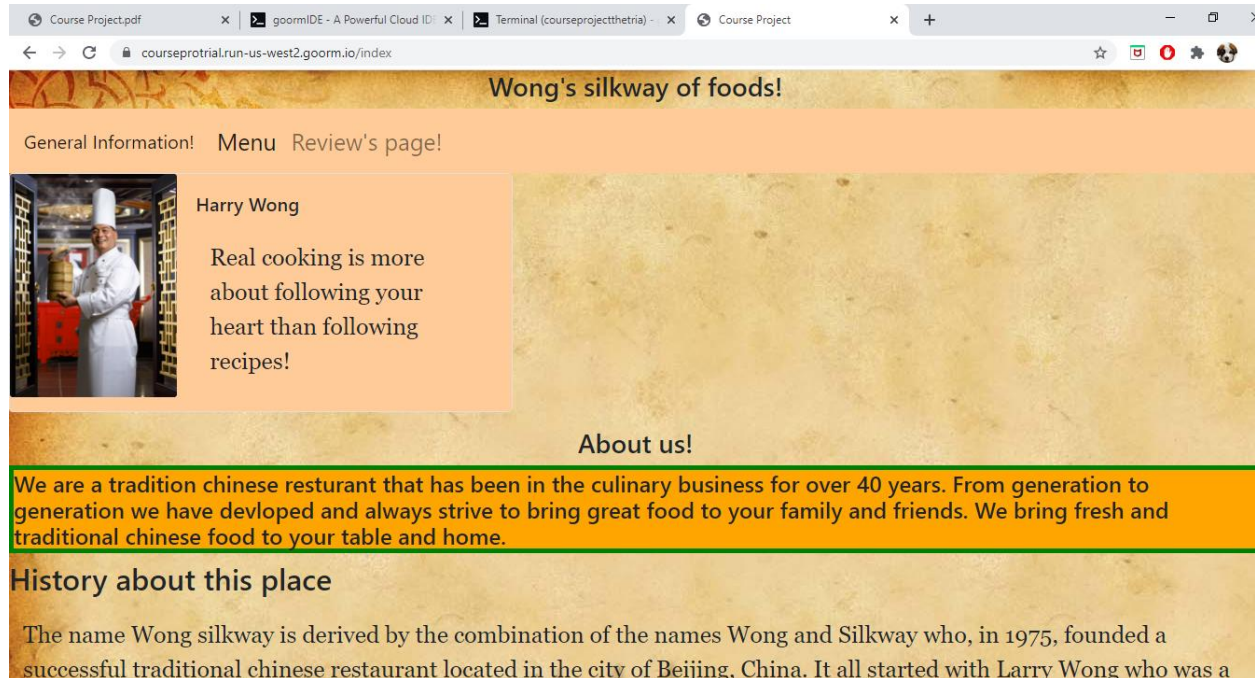
Description: hospitality industry, take-out food.

Compare Figma Wireframes with the Screenshots of your website



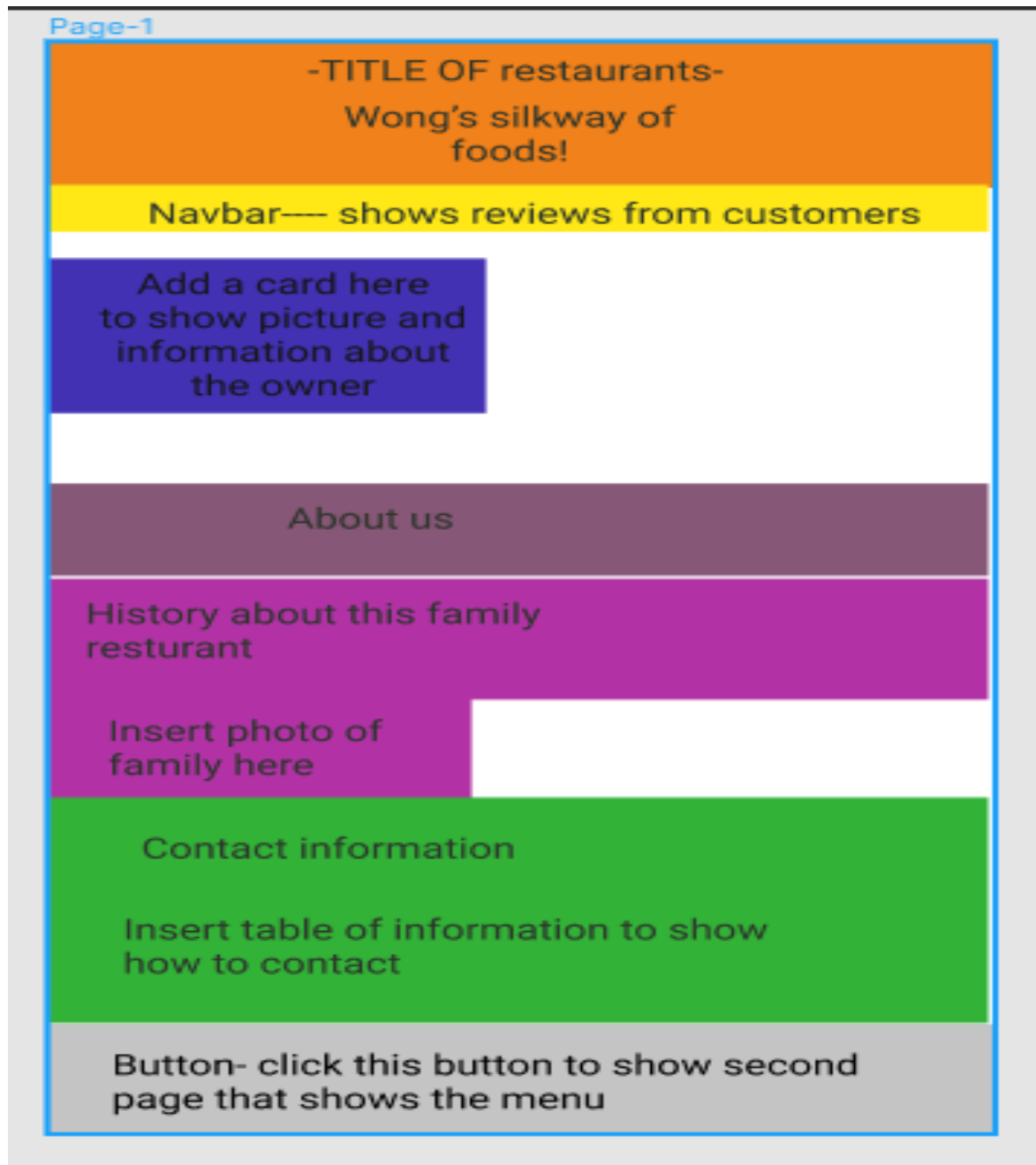
I have included another 2 pages. Both are login/signup pages to access into my website. These 2 pages are basically created for the backend part of my website. So people will be authorized to access it.

Page 1 of my website



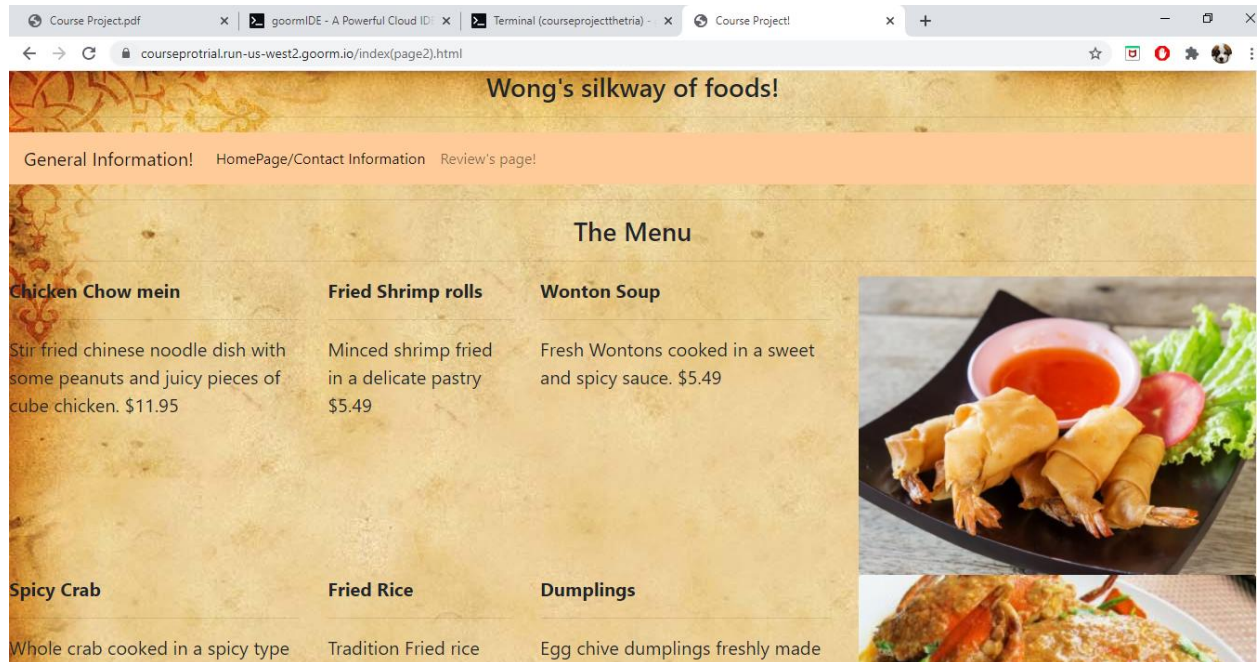
As well as that. The colors on my figma designs are different from my website designs since I felt like to many colours would not make the website really appealing.

As well as I have added some extra features into the navbar as well. (I explained more in detail down below)



The only difference is that I have edited my navbar to be able to access the other page using the navbar as well. As well as I added a log out & sign in feature on the bottom of the first page as well.

Page 2 of my website



-Title of resturant

Wong's silkway of foods!

Menu

column 1

Column 2

Column 3

Pictures of some of the food items will be posted on this side

Collapse button

Add one here to recommend some dishes

Covid-19 special combo menu

Description of the 3 combos.

Carousel of pictures of the 3 combo items

Button- click this button to show first page that shows the main information

The only difference is that I have edited my navbar to be able to access the other page using the navbar as well.

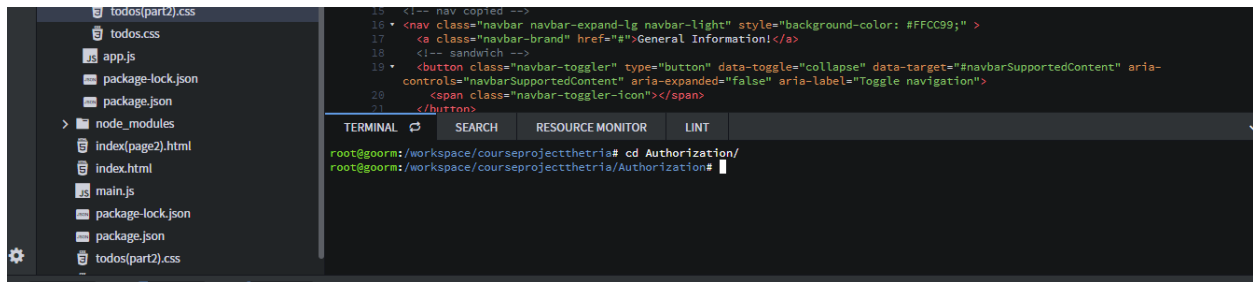
Instructions to run your project in GoormIDE

Step 1: First click the goormide link I have provided in my submission.

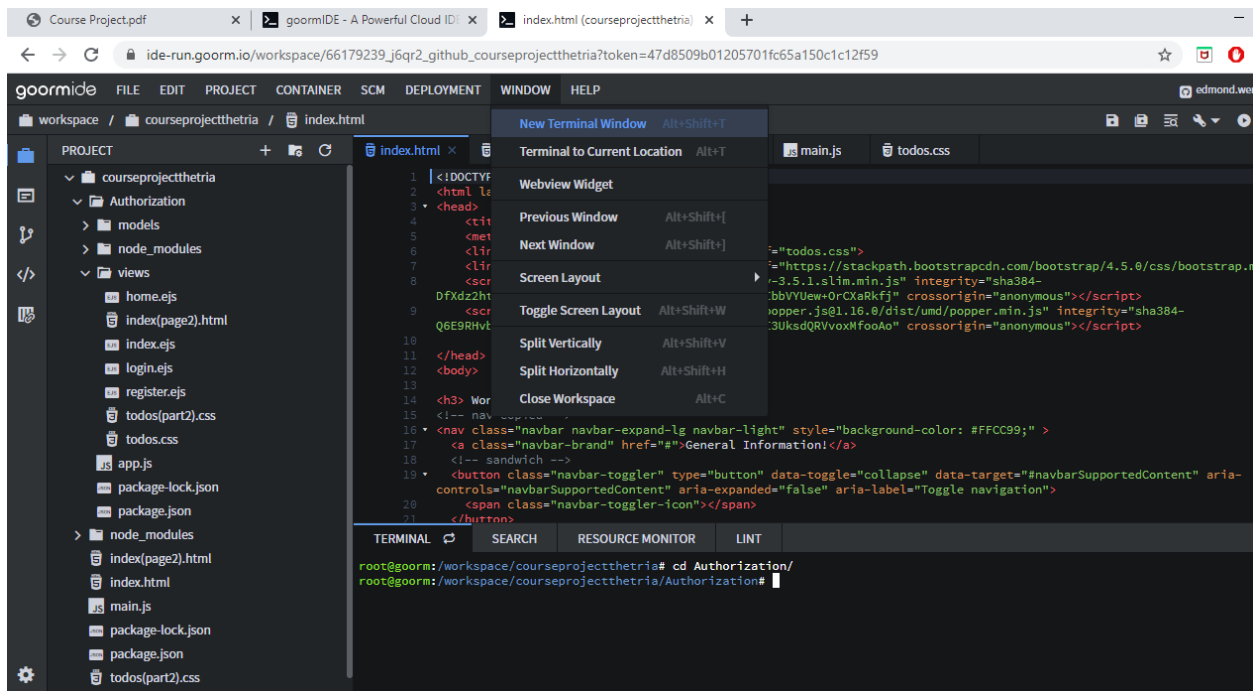
(Here is the link again if you have not accessed goormide yet.

<https://goor.me/7NL6Y>)

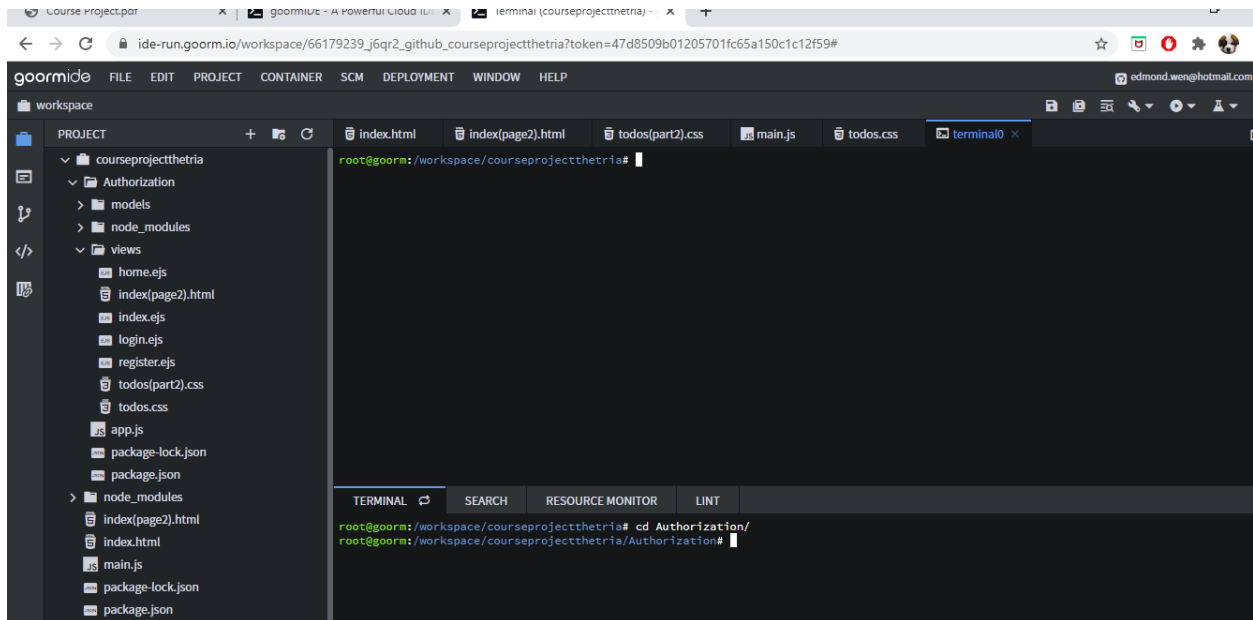
Step 2: Type in “cd Authorization/” into the terminal. Or type “cd A” and press tab for short cut.



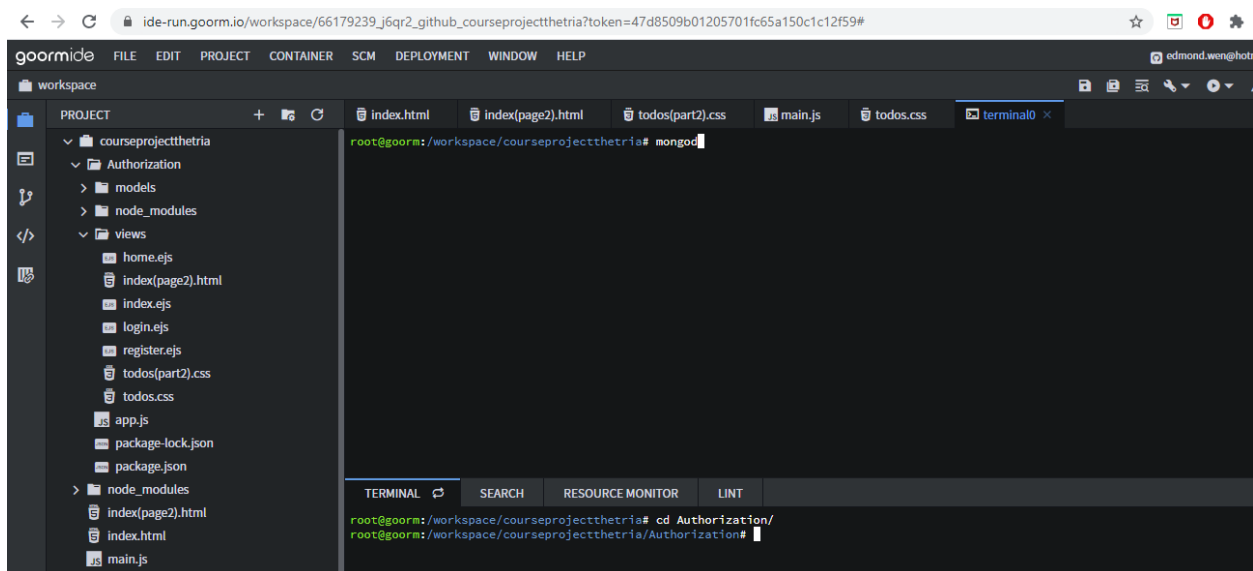
Step 3: Select window → and then create a new terminal window(like the screenshot below)



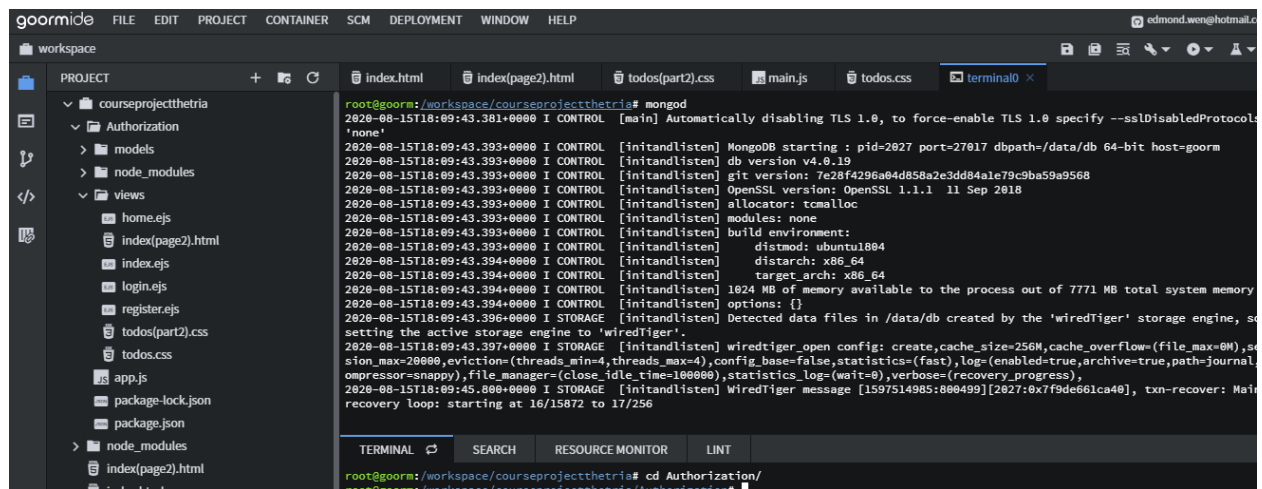
Step 4: The new terminal should look like the screenshot.



Step 5: Type in “mongod” into the top terminal (like the screenshot)



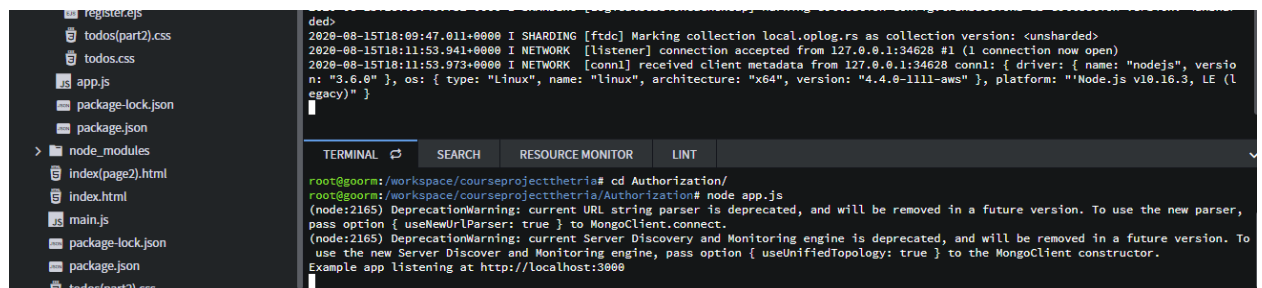
And then press enter and wait a second.



The screenshot shows the Goorm IDE interface. The left sidebar displays a project tree for 'courseprojectthetria' with folders like 'Authorization', 'models', 'node_modules', and 'views'. The main editor area shows a terminal window with the following output:

```
root@goorm:/workspace/courseprojectthetria# mongod
2020-08-15T18:09:43.381+0000 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] MongoDB starting : pid=2027 port=27017 dbpath=/data/db 64-bit host=goorm
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] db version v4.0.19
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] git version: 7e28f4296a8d858a2e3dd84a1e79c9ba59a5568
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] OpenSSL version: OpenSSL 1.1.1 11 Sep 2018
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] allocator: tcmalloc
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] modules: none
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] build environment:
2020-08-15T18:09:43.393+0000 I CONTROL [initandlisten] distmod: ubuntu1804
2020-08-15T18:09:43.394+0000 I CONTROL [initandlisten] distarch: x86_64
2020-08-15T18:09:43.394+0000 I CONTROL [initandlisten] target_arch: x86_64
2020-08-15T18:09:43.394+0000 I CONTROL [initandlisten] 1024 MB of memory available to the process out of 7771 MB total system memory
2020-08-15T18:09:43.396+0000 I STORAGE [initandlisten] Detected data files in /data/db created by the 'wiredTiger' storage engine, so
2020-08-15T18:09:43.397+0000 I STORAGE [initandlisten] setting the active storage engine to 'wiredTiger'.
2020-08-15T18:09:43.397+0000 I STORAGE [initandlisten] wiredtiger_open config: create,cache_size=256M,cache_overflow=(file_max=6M),se
2020-08-15T18:09:43.397+0000 I STORAGE [initandlisten] session_max=20000,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,
2020-08-15T18:09:43.397+0000 I STORAGE [initandlisten] compressor=snappy),file_manager=(close_idle_time=100000),statistics_log=(wait=0),verbose=(recovery_progress),
2020-08-15T18:09:45.800+0000 I STORAGE [initandlisten] WiredTiger message [1597514985:800499][2027:0x7f9de601ca40]: txn-recover: Main
2020-08-15T18:09:45.800+0000 I STORAGE [initandlisten] recovery loop: starting at 16/15872 to 17/256
```

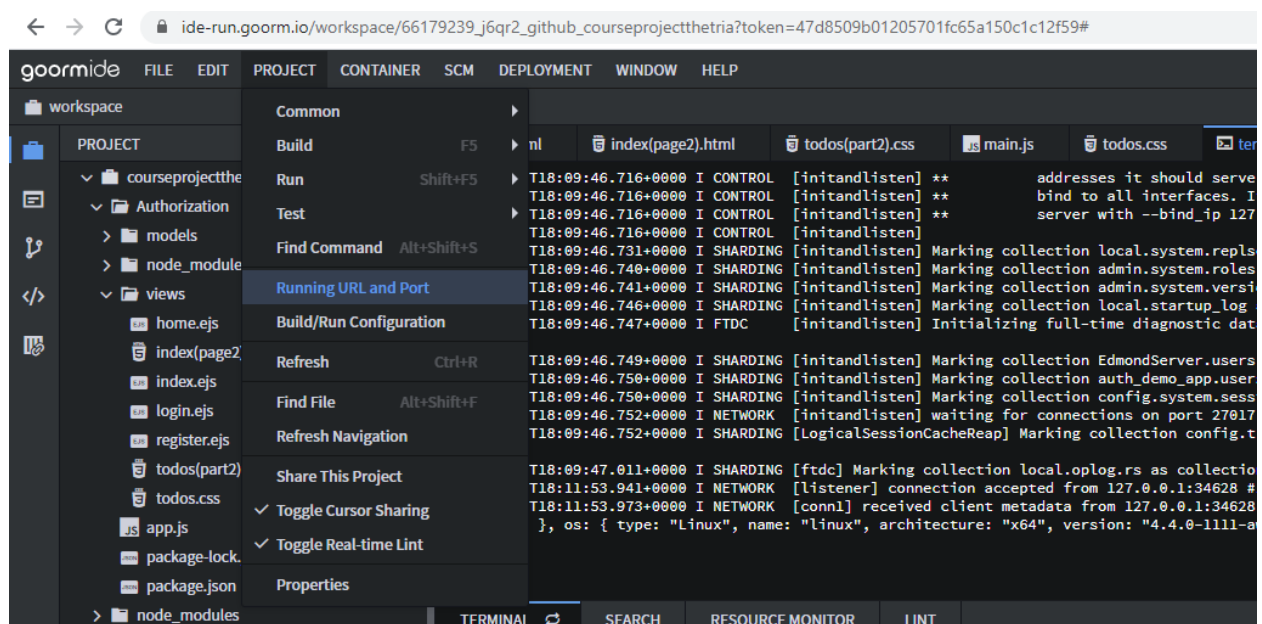
Step 6: Then go back to the terminal at the bottom and then type in “node app.js”



The screenshot shows the Goorm IDE interface. The left sidebar displays a project tree for 'courseprojectthetria' with folders like 'Authorization', 'models', 'node_modules', and 'views'. The main editor area shows a terminal window with the following output:

```
root@goorm:/workspace/courseprojectthetria# cd Authorization/
root@goorm:/workspace/courseprojectthetria/Authorization# node app.js
(node:2165) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser,
pass option { useNewUrlParser: true } to MongoClient.connect.
(node:2165) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To
use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Example app listening at http://localhost:3000
```

Step 7: Go to project and the running and URL and Port.



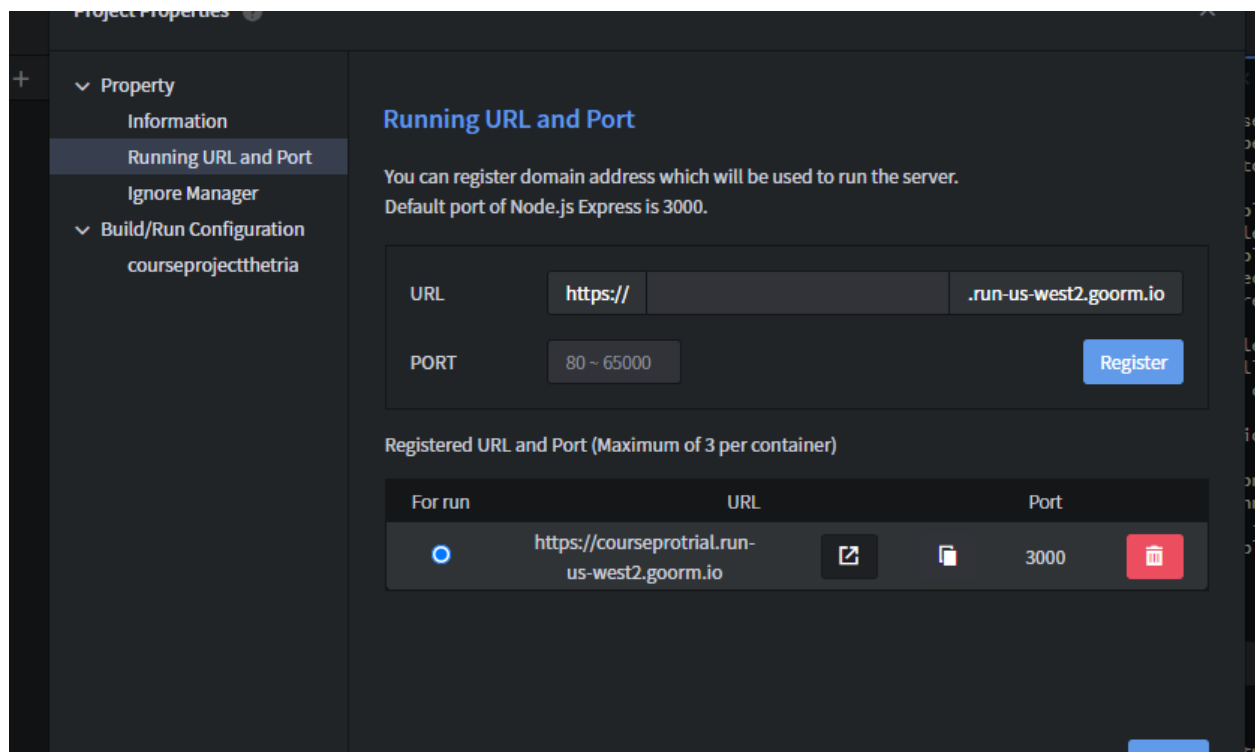
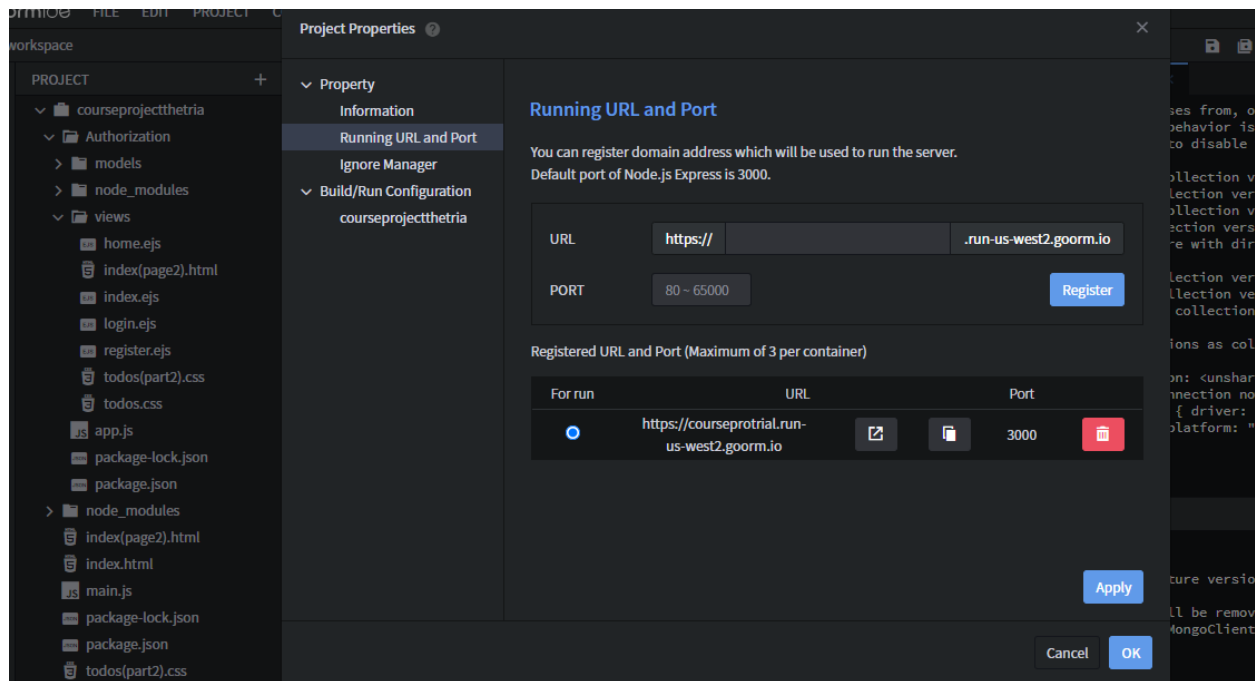
The screenshot shows the Goorm IDE interface. The left sidebar displays a project tree for 'courseprojectthetria' with folders like 'Authorization', 'models', 'node_modules', and 'views'. The main editor area shows a terminal window with the following output:

```
root@goorm:/workspace/courseprojectthetria# cd Authorization/
root@goorm:/workspace/courseprojectthetria/Authorization# node app.js
(node:2165) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser,
pass option { useNewUrlParser: true } to MongoClient.connect.
(node:2165) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To
use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Example app listening at http://localhost:3000
```

The right sidebar shows a list of actions for the 'Authorization' project. The 'Running URL and Port' option is highlighted. The output of this action is displayed in the terminal window:

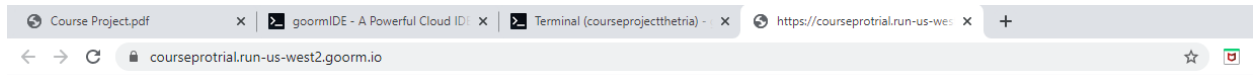
```
T18:09:46.716+0000 I CONTROL [initandlisten] ** addresses it should serve
T18:09:46.716+0000 I CONTROL [initandlisten] ** bind to all interfaces. I
T18:09:46.716+0000 I CONTROL [initandlisten] ** server with --bind_ip 127
T18:09:46.716+0000 I CONTROL [initandlisten]
T18:09:46.731+0000 I SHARDING [initandlisten] Marking collection local.system.repl
T18:09:46.740+0000 I SHARDING [initandlisten] Marking collection admin.system.roles
T18:09:46.741+0000 I SHARDING [initandlisten] Marking collection admin.system.version
T18:09:46.746+0000 I SHARDING [initandlisten] Marking collection local.startup_log
T18:09:46.747+0000 I FTDC [initandlisten] Initializing full-time diagnostic data
```

Step 8: Either press the website tab or copy the link and paste into a new tab.



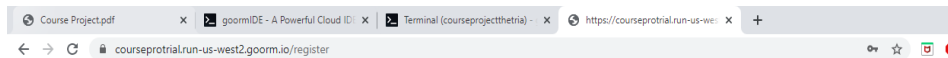
<https://courseprotrial.run-us-west2.goorm.io>

Step 9: Create a sign-up account and password (anything works) and the you should be able to connect.



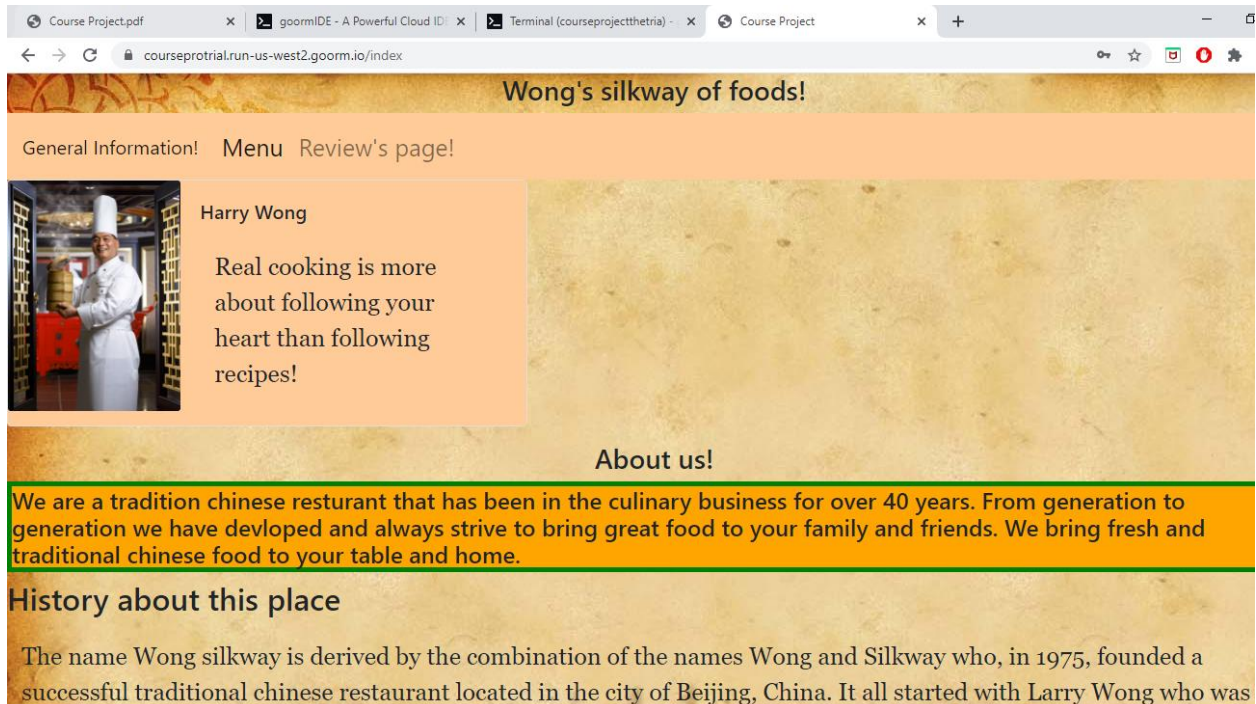
Home

- [Sign Up!](#)
- [Login](#)
- [Logout](#)



Sign Up Form

Step 10: You should be able to access my website.



If you log out by accident from below.



You can login again with the same account you have created previously. By using the login feature.

Feature/Content Explanations

For the customer's/ business perspective, I set up the website in a way where it is like journey for them to see. The first page will be information about the restaurant and the types of cuisines that the restaurant will host. Also, it would talk about the history about the restaurant as well. It will also include contact information to book take out meals (through the phone) and other types of information.

On the second page of the website it would include the menu and prices of the foods they can order using a phone call. Also, on the second page it will have pictures that shows what the food looks like that would attract customers. As well as huge special combos that has pictures that are being represented with a carousel. Customers likes pictures of food and deals so they would likely spend more.

As for the features of the website. I would add a card to show a picture of the owner of the restaurant and his quotes of why we should eat there. As well as a navbar to show reviews of previous customers that have ate at the restaurants. I would insert tables to make the website cleaner. As well as buttons on BOTH pages to connect customers to switch between the pages however they choose to. Especially since they need to order online and use the contact information and the menu. On the second page I would have 4 columns. Columns 1-3 are all menu items plus the price of the food and the last column will be photo of one of the items on the menu. There will be a carousel and a collapse button to help make customers have more information and such.

(explanation of features and more VVV)

I have added some features like collapse buttons and navbars to make it easier for the customer or whoever is using the website. To make it more convenient for

users to access. Since the navbar connects both pages and the carousel help show the foods without having 3 pictures on the website. I have set up the menu in a simple type of style to create an effect of easy going for the customer as well. As well as contact information in the front.

```

Kinds of dishes that people have always enjoyed for awhile now. These combos have to be requested ahead of time so our professional chefs time to prepare them.</p>
<div class="row">
  <div class="col-sm-4"></div>
  <div class="col-sm-4">
    <div id="carouselExampleFade" class="carousel slide carousel-fade" data-ride="carousel">
      <div class="carousel-inner">
        <div class="carousel-item active" data-interval="3000">
          
        </div>
        <div class="carousel-item" data-interval="3000">
          
        </div>
        <div class="carousel-item" data-interval="3000">
          
        </div>
      </div>
    </div>
  </div>
</div>
</div>
</div>

```

(carousel ^^^^)

As well I have added a sign in /login feature so users would feel safer accessing the website. To log out of your account you can press the log out on the first page. The navbar connects both websites as well as show yelp reviews of the restaurant as well. Also, I set the menu into 4 parts of the grid system. The first 3 parts are menu items and the 4th are just a picture of one of the three items that was shown. So, it would make the food more appeal for customers.