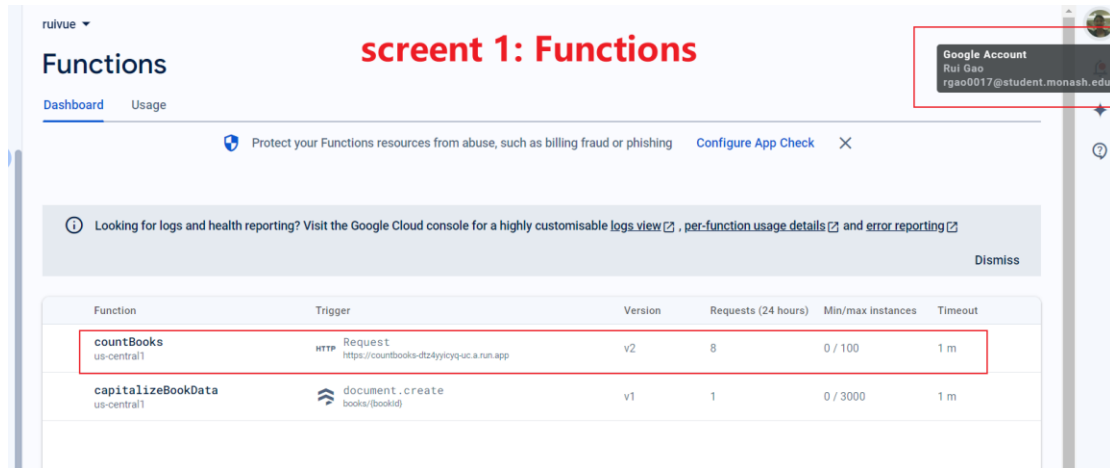


Task 9.1



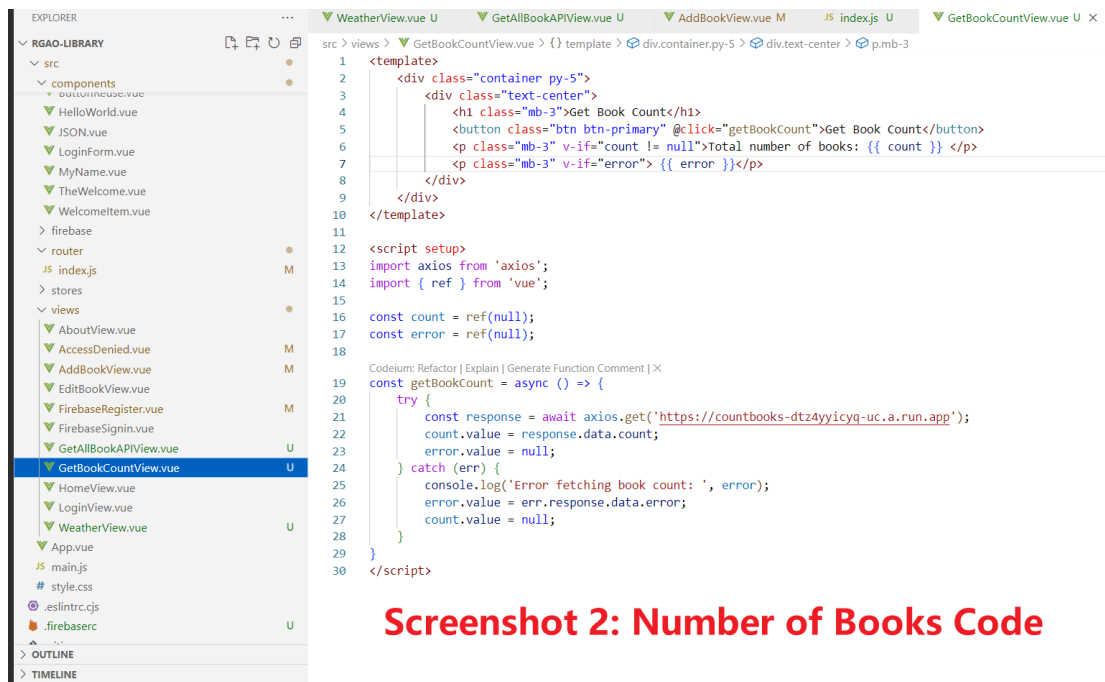
Home (Week 5) Get Weather **Get Book Count** Get All Book API Add Book Edit Book About Logout

Get Book Count

Get Book Count

Total number of books: 5

Screenshot 2: Number of Books



Screenshot 2: Number of Books Code

Task 9.2

[Home \(Week 5\)](#) [Get Weather](#) [Get Book Count](#) [Get All Book API](#) [Add Book](#) [Edit Book](#) [About](#) [Logout](#)

Add Book

ISBN:

eg. 9783161484100

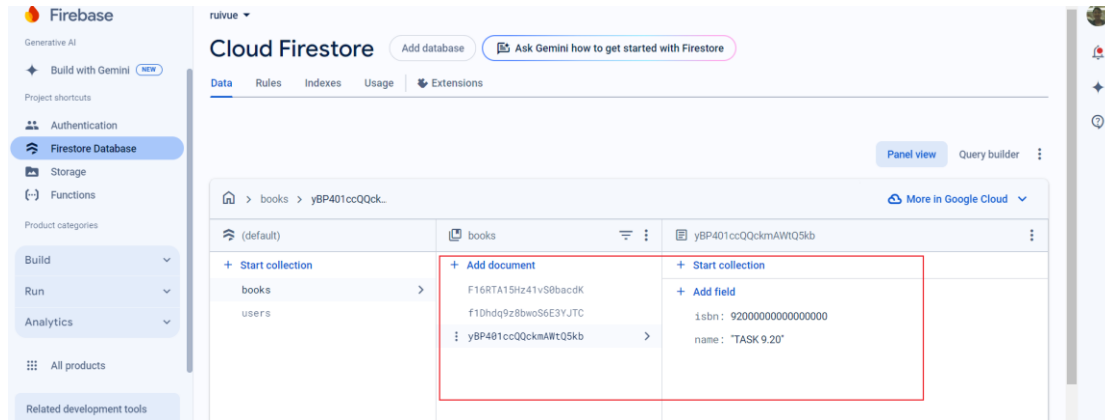
Name:

Enter name of the book

[Add Book](#)

Added Books

ISBN	Name
82347927	A GOOD BOOK
11111	A BAD BOOK
9200000000000000000	TASK 9.20



```
src > views > AddBookView.vue > {} template > div.container > div.row
You, 4 minutes ago | 1 author (You)

1 <template>
2   <div class="container">
3     <div class="row">
4       <div class="col-md-6 offset-md-3">
5         <form @submit.prevent="addBook" class="flex">
6           <h1 class="text-center my-5">Add Book</h1>
7           <div class="form-group mb-3">
8             <label for="isbn">ISBN:</label>
9             <input v-model="isbn" id="isbn" type="text" class="form-control" placeholder="eg. 9783161484100"
10              required />
11           </div>
12           <div class="form-group mb-3">
13             <label for="name">Name:</label>
14             <input v-model="name" id="name" type="text" class="form-control"
15              placeholder="Enter name of the book" required />
16           </div>
17           <button class="btn btn-primary" type="submit">Add Book</button>
18         </form>
19       </div>
20     </div>
21     <div class="row">
22       <div class="col-md-8 offset-md-2">
23         <h1 class="text-center mt-5 mb-3">Added Books</h1>
24         <table class="table">
25           <thead class="table-light">
26             <tr>
27               <th>ISBN</th>
28               <th>Name</th>
29             </tr>
30           </thead>
31           <tbody>
32             <tr v-for="book in books" :key="book.isbn">
33               <td>{{ book.isbn }}</td>
34               <td>{{ book.name }}</td>
35             </tr>
36           </tbody>
        </table>
      </div>
    </div>
  </div>
</template>
```

```

    </table>
  </div>
</div>
</div>
</template>

```

```

<script setup>
import { onMounted, ref } from 'vue'
import db from '@/firebase/init'
import { collection, addDoc, getDocs } from 'firebase/firestore'

const books = ref([]);
const isbn = ref('')
const name = ref('')

```

Codeium: Refactor | Explain | Generate Function Comment | X

```

const addBook = async () => {
  try {
    const isbnNumber = Number(isbn.value)
    if (isNaN(isbnNumber)) {
      alert('ISBN must be a valid number')
      return
    }

    await addDoc(collection(db, 'books'), {
      isbn: isbnNumber,
      name: name.value
    })

    isbn.value = ''
    name.value = ''
    updateBookList()
    alert('Book added successfully')
  } catch (error) {
    console.error('Error adding book: ', error)
  }
}

```

Codeium: Refactor | Explain | Generate Function Comment | X

```

const updateBookList = async () => {
  const booksCollection = collection(db, "books");
  const booksSnapshot = await getDocs(booksCollection);
  books.value = booksSnapshot.docs.map((doc) => ({ id: doc.id, ...doc.data() }));
};

onMounted(() => {
  updateBookList();
});

</script>

```

Task 10.1

Home (Week 5) **Get Weather** Get Book Count Count Book API Get All Book API Add Book Edit Book About Logout

screenshot 1: current location weather

WEATHER APP

Enter city name Search

Richmond, AU



15 °C

overcast clouds

```
WeatherView.vue U x GetAllBookAPIView.vue U AddBookView.vue M JS index.js U GetBookCountView.vue U
src > views > WeatherView.vue > {} script setup > [0] fetchWeatherData
1 <template>
2   <div class="container py-5">
3     <div class="header text-center">
4       <h1 class="mb-3">WEATHER APP</h1>
5       <div class="search-bar mb-3">
6         <input type="text" v-model="city" placeholder="Enter city name" class="search-input" />
7         <button @click="searchByCity" class="search-button">Search</button>
8       </div>
9     </div>
10    <main class="text-center">
11      <div v-if="weatherData">
12        <h2>
13          {{ weatherData.name }}, {{ weatherData.sys.country }}
14        </h2>
15        <div>
16          
17          <p>{{ temperature }} °C</p>
18        </div>
19        <span>{{ weatherData.weather[0].description }}</span>
20      </div>
21    </main>
22  </div>
23 </template>
24
25 <script setup>
26 import axios from "axios";
27 import { ref, computed, onMounted } from "vue";
28
29 const apiKey = "be44c19be5c3a4e5f4ad3ae2bb72dcf7";
30 const city = ref("");
31 const weatherData = ref(null);
32
33 const temperature = computed(() => (
34   weatherData.value ? Math.floor(weatherData.value.main.temp - 273) : null
35 ));
36
```

Get weather code

```

const temperature = computed(() => (
  weatherData.value ? Math.floor(weatherData.value.main.temp - 273) : null
));

const iconUrl = computed(() => (
  weatherData.value ? `http://api.openweathermap.org/img/w/${weatherData.value.weather[0].icon}.png` : null
));

Codeium: Refactor | Explain | Generate Function Comment | X
const fetchCurrentLocationWeather = async () => {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(async position => {
      const { latitude, longitude } = position.coords;
      const url = `https://api.openweathermap.org/data/2.5/weather?lat=${latitude}&lon=${longitude}&appid=${apikey}`;
      await fetchWeatherData(url);
    });
  }
}

Codeium: Refactor | Explain | Generate Function Comment | X
const fetchWeatherData = async (url) => {
  try {
    const response = await axios.get(url);
    weatherData.value = response.data;
  } catch (error) {}
  console.error("Error fetching weather data:", error);
}

Codeium: Refactor | Explain | Generate Function Comment | X
const searchByCity = async () => {
  const url = `https://api.openweathermap.org/data/2.5/weather?q=${city.value}&appid=${apikey}`;
  await fetchWeatherData(url);
}

onMounted(() => {
  fetchCurrentLocationWeather();
});
</script>

```

← → ↻ ① localhost:5173/CountBookAPI

🗑️ | 📁 Monash 📁 单词学习网站 📁 英语电视节目 📁 英语听力 📁 老师推荐网站 📁 学习 📧 Gmail

```

{
  "count": 3
}

```

Screenshot 2: API page counters

```
App.vue M GetAllBookAPI.vue U X
src > views > GetAllBookAPI.vue > {} script setup > books
1 <template>
2 <!-- <div class="container py-5">
3 <h1>All Books (JSON Format)</h1>
4 <hr />
5 <pre>{{ books }}</pre>
6 </div> -->
7 <pre>{{ books }}</pre>
8 </template>
9
10 <script setup>
11 import { onMounted, ref } from "vue";
12 import { collection, getDocs } from "firebase/firestore";
13 import db from '@/firebase/init';
14
15
16 const books = ref([]);
17
18 Codeium: Refactor | Explain | Generate Function Comment | X
19 const getAllBooks = async () => {
20   const booksCollection = collection(db, "books");
21   const booksSnapshot = await getDocs(booksCollection);
22   const booksList = booksSnapshot.docs.map((doc) => ({ id: doc.id, ...doc.data() }));
23   return booksList;
24 };
25
26 onMounted(async () => {
27   books.value = await getAllBooks(); // Fetch all books on component mount
28 });
29
30 </script>
```

Screenshot 2: API page code

Task 10.2

Home (Week 5) Get Weather Get Book Count Count Book API Get All Book API Add Book Edit Book About Logout

screenshot 1: search city weather

WEATHER APP

clayton, AU Search

Clayton, AU



14 °C

broken clouds

Codeium: Refactor | Explain | Generate Function Comment | ✕

```
const searchByCity = async () => {  
  const url = `https://api.openweathermap.org/data/2.5/weather?q=${city.value}&appid=${apikey}`;  
  await fetchWeatherData(url);  
}  
  
onMounted(() => {  
  fetchCurrentLocationWeather();  
});  
</script>
```


search city weather

Home (Week 5) **Get Weather** Get Book Count Count Book API **Get All Book API** Add Book Edit Book About Logout

screenshot 2: get all book api

WEATHER APP

Richmond, AU



15 °C

overcast clouds

← → ↺ 🌐 localhost:5173/GetAllBookAPI

📖 | 📁 Monash 📁 单词学习网站 📁 英语电视节目 📁 英语听力 📁 老师推荐网站 📁 学习 📧 Gmail

```
[  
  {  
    "id": "F16RTA15Hz41vS0bacdk",  
    "name": "A GOOD BOOK",  
    "isbn": 82347927  
  },  
  {  
    "id": "f1Dhdq9z8bwo56E3VJTC",  
    "name": "A BAD BOOK",  
    "isbn": 11111  
  },  
  {  
    "id": "yBP401ccQQckmAw05kb",  
    "isbn": 9200000000000000,  
    "name": "TASK 9.20"  
  }  
]
```

1

screenshot 2: get all book api

App.vue MGetAllBookAPI.vue UWeatherView.vue U

src > views > GetAllBookAPI.vue > {} script setup > books

```
1 <template>
2   <!-- <div class="container py-5">
3     <h1>All Books (JSON Format)</h1>
4     <hr />
5     <pre>{{ books }}</pre>
6   </div> -->
7   <pre>{{ books }}</pre>
8 </template>
9
10 <script setup>
11   import { onMounted, ref } from "vue";
12   import { collection, getDocs } from "firebase/firestore";
13   import db from '@/firebase/init';
14
15
16   const books = ref([]);
17
18   Codeium: Refactor | Explain | Generate Function Comment | X
19   const getAllBooks = async () => {
20     const booksCollection = collection(db, "books");
21     const booksSnapshot = await getDocs(booksCollection);
22     const booksList = booksSnapshot.docs.map((doc) => ({ id: doc.id, ...doc.data() }));
23     return booksList;
24   };
25
26   onMounted(async () => {
27     books.value = await getAllBooks(); // Fetch all books on component mount
28   });
29
30 </script>
```

get all books api code