

# [CGA-JAVA-WFDA] Web Front-end Development Angular 2.1

[Dashboard](#) / [My courses](#) / [CGA-JAVA-WFDA-2.1](#) / [9. Kết nối với Back-end](#) / [\[Thực hành\] Xây dựng ứng dụng quản lý sản phẩm](#)

## [Thực hành] Xây dựng ứng dụng quản lý sản phẩm

### Mục tiêu

Sử dụng HttpClient trong Angular

### Mô tả

Sử dụng ứng dụng quản lý sản phẩm đã được xây dựng ở bài Module tại [đây](#) và sử dụng HttpClientModule để gọi API từ bên phía backend

### Yêu cầu

Ứng dụng quản lý sản phẩm phải được xây dựng như sau:

Sử dụng API bên phía backend để thực hiện thao tác CRUD cho Category

Sử dụng các phương thức cơ bản của HttpClient

### Hướng dẫn

Bước 1: Khai báo HttpClientModule trong file app.module.ts như sau:

```
import {BrowserModule} from '@angular/platform-browser';
import {NgModule} from '@angular/core';

import {AppRoutingModule} from './app-routing.module';
import {AppComponent} from './app.component';
import {SharedModule} from './shared/shared.module';
import {HttpClientModule} from '@angular/common/http';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    SharedModule,
    HttpClientModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule {
}
```

Bước 2: Mở file enviroment.ts trong thư mục enviroments và khai báo thuộc tính apiUrl để cấu hình đường dẫn resource của backend

```
export const environment = {
  production: false,
```

```
    apiUrl: 'http://localhost:8080'
  };
```

Bước 3: Xóa bỏ khai báo danh sách danh mục sản phẩm trong file category.service.ts và khai báo thêm một thuộc tính HttpClient trong constructor như sau:

```
const API_URL = `${environment.apiUrl}`;
@Injectable({
  providedIn: 'root'
})
export class CategoryService {

  constructor(private http: HttpClient) {

  }

  ...

}
```

Bước 4: Cấu hình lại phương thức getAll() với kiểu trả về là một Observable

```
getAll(): Observable<Category[]> {
  return this.http.get<Category[]>(API_URL + '/categories');
}
```

Bước 5: Sau đó chỉnh sửa lần lượt các phương thức saveCategory, updateCategory, findById và deleteCategory ta được category.service.ts mới như sau:

```
import {Injectable} from '@angular/core';
import {Category} from '../model/category';
import {HttpClient} from '@angular/common/http';
import {Observable} from 'rxjs';
import {environment} from '../../environments/environment';

const API_URL = `${environment.apiUrl}`;

@Injectable({
  providedIn: 'root'
})
export class CategoryService {

  constructor(private http: HttpClient) {
  }

  getAll(): Observable<Category[]> {
    return this.http.get<Category[]>(API_URL + '/categories');
  }

  saveCategory(category): Observable<Category> {
    return this.http.post<Category>(API_URL + '/categories', category);
  }

  findById(id: number): Observable<Category> {
    return this.http.get<Category>(`${API_URL}/categories/${id}`);
  }

  updateCategory(id: number, category: Category): Observable<Category> {
    return this.http.put<Category>(`${API_URL}/categories/${id}`, category);
  }

  deleteCategory(id: number): Observable<Category> {
    return this.http.delete<Category>(`${API_URL}/categories/${id}`);
  }
}
```

Bước 6: Chỉnh sửa lại các component của Category thành như sau:

category-list.component.ts

```
import {Component, OnInit} from '@angular/core';
import {Category} from '../..../model/category';
import {CategoryService} from '../..../service/category.service';

@Component({
  selector: 'app-category-list',
  templateUrl: './category-list.component.html',
  styleUrls: ['./category-list.component.css']
})
export class CategoryListComponent implements OnInit {
  categories: Category[] = [];

  constructor(private categoryService: CategoryService) {
  }

  ngOnInit() {
    this.getAll();
  }

  getAll() {
    this.categoryService.getAll().subscribe(categories => {
      this.categories = categories;
    });
  }
}
```

category-create.component.ts

```
import {Component, OnInit} from '@angular/core';
import {FormControl, FormGroup} from '@angular/forms';
import {CategoryService} from '../../service/category.service';

@Component({
  selector: 'app-category-create',
  templateUrl: './category-create.component.html',
  styleUrls: ['./category-create.component.css']
})
export class CategoryCreateComponent implements OnInit {
  categoryForm: FormGroup = new FormGroup({
    name: new FormControl(),
  });

  constructor(private categoryService: CategoryService) {
  }

  ngOnInit() {
  }

  submit() {
    const category = this.categoryForm.value;
    this.categoryService.saveCategory(category).subscribe(() => {
      this.categoryForm.reset();
      alert('Tạo thành công');
    }, e => {
      console.log(e);
    });
  }
}
```

category-create.component.html

```
<h1>Tạo danh mục mới</h1>
<form [formGroup]="categoryForm" (ngSubmit)="submit()">
  <table>
    <tr>
      <td>Tên danh mục:</td>
      <td><input type="text" formControlName="name"></td>
    </tr>
    <tr>
      <td></td>
      <td>
        <button>Tạo mới</button>
      </td>
    </tr>
  </table>
```

```
</table>
</form>
```

category-edit.component.ts

```
import {Component, OnInit} from '@angular/core';
import {FormControl, FormGroup} from '@angular/forms';
import {ActivatedRoute, ParamMap} from '@angular/router';
import {CategoryService} from '../../service/category.service';

@Component({
  selector: 'app-category-edit',
  templateUrl: './category-edit.component.html',
  styleUrls: ['./category-edit.component.css']
})
export class CategoryEditComponent implements OnInit {
  categoryForm: FormGroup;
  id: number;

  constructor(private categoryService: CategoryService,
               private activatedRoute: ActivatedRoute) {
    this.activatedRoute.paramMap.subscribe((paramMap: ParamMap) => {
      this.id = +paramMap.get('id');
      this.getCategory(this.id);
    });
  }

  ngOnInit() {
  }

  getCategory(id: number) {
    return this.categoryService.findById(id).subscribe(category => {
      this.categoryForm = new FormGroup({
        name: new FormControl(category.name),
      });
    });
  }

  updateCategory(id: number) {
    const category = this.categoryForm.value;
    this.categoryService.updateCategory(id, category).subscribe(() => {
      alert('Cập nhật thành công');
    }, e => {
      console.log(e);
    });
  }
}
```

category-edit.component.html



```
<h1>Cập nhật thông tin danh mục sản phẩm</h1>
<form [formGroup]="categoryForm" (ngSubmit)="updateCategory(id)">
  <table>
    <tr>
      <td>Tên danh mục:</td>
      <td><input type="text" formControlName="name"></td>
    </tr>
    <tr>
      <td></td>
      <td><button>Cập nhật</button></td>
    </tr>
  </table>
</form>
```

category-delete.component.ts

```
import {Component, OnInit} from '@angular/core';
import {FormControl, FormGroup} from '@angular/forms';
import {CategoryService} from '../../service/category.service';
import {ActivatedRoute, ParamMap, Router} from '@angular/router';

@Component({
  selector: 'app-category-delete',
  templateUrl: './category-delete.component.html',
  styleUrls: ['./category-delete.component.css']
})
export class CategoryDeleteComponent implements OnInit {
  categoryForm: FormGroup;
  id: number;

  constructor(private categoryService: CategoryService,
               private router: Router,
               private activatedRoute: ActivatedRoute) {
    this.activatedRoute.paramMap.subscribe((paramMap: ParamMap) => {
      this.id = +paramMap.get('id');
      this.getCategory(this.id);
    });
  }

  ngOnInit() {
  }

  getCategory(id: number) {
    return this.categoryService.findById(id).subscribe(category => {
      this.categoryForm = new FormGroup({
        name: new FormControl(category.name),
      });
    });
  }

  deleteCategory(id: number) {
    this.categoryService.deleteCategory(id).subscribe(() => {
      this.router.navigate(['/category/list']);
    }, e => {
      console.log(e);
    });
  }
}
```

category-delete.component.html



```
<h1>Xoá danh mục sản phẩm</h1>
<form [formGroup]="categoryForm" (ngSubmit)="deleteCategory(id)">
  <table>
    <tr>
      <td>Tên danh mục:</td>
      <td><input type="text" formControlName="name"></td>
    </tr>
    <tr>
      <td></td>
      <td><button>Xoá</button></td>
    </tr>
  </table>
</form>
```

Bước 7: Chạy thử chương trình và kiểm tra kết quả

Checkout sang nhánh connect-backend để xem mã nguồn tham khảo: <https://github.com/codegym-vn/wfda-angular-product-management>

## Hướng dẫn nộp bài:

up bài tập lên github

paste link github vào phần nộp bài

### Submission status

Submission status	No attempt
Grading status	Not graded
Last modified	-
Submission comments	<div>▶ <a href="#">Comments(0)</a></div>

Add submission

You have not made a submission yet.

Previous Activity

Jump to...

Next Activity

#### CHƯƠNG TRÌNH

- Career
- Premium
- Accelerator

#### TÀI NGUYÊN

- Blog
- Tạp chí Lập trình
- AgileBreakfast

Follow Us



CodeGym@2018. All rights reserved.

You are logged in as Dương Văn Thanh Sơn (Log out)

