

Tạo ra ProductCategoryViewModel chứa các thuộc tính của ProductCategory

Nhớ thêm Id vào (trong entity Id thừa kế từ DomainEntity nên ko có)

Tách model ra không để có thể thêm bớt thuộc tính, gộp model mà không bị phụ thuộc entity

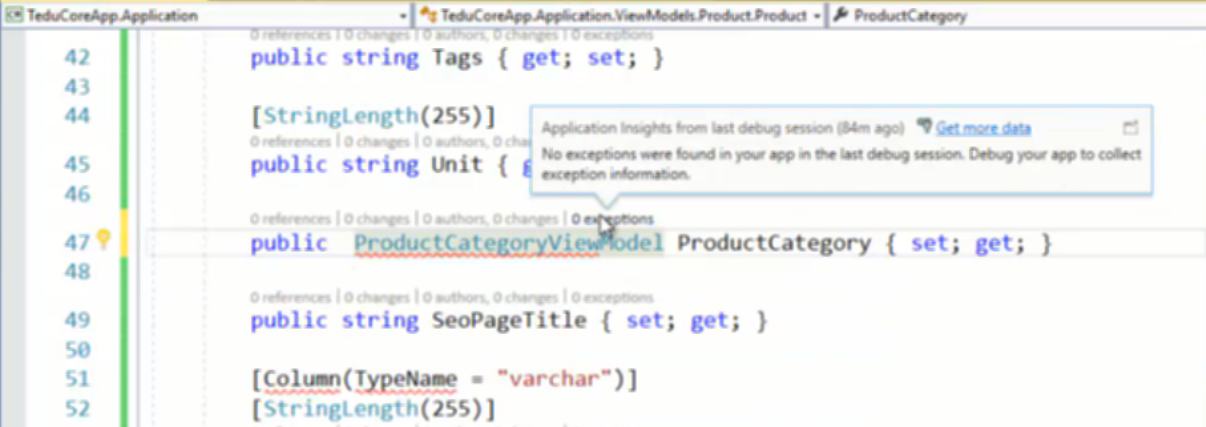




ở đây không dùng để query nên bỏ virtual đi

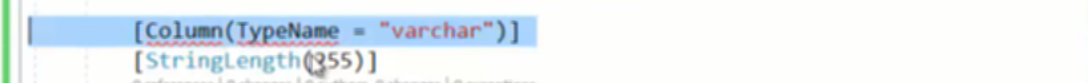
và đổi thành ProductViewModel

Tương tự ProductCategory-> với product tạo ProductViewModel



Bỏ ForeinKey đi

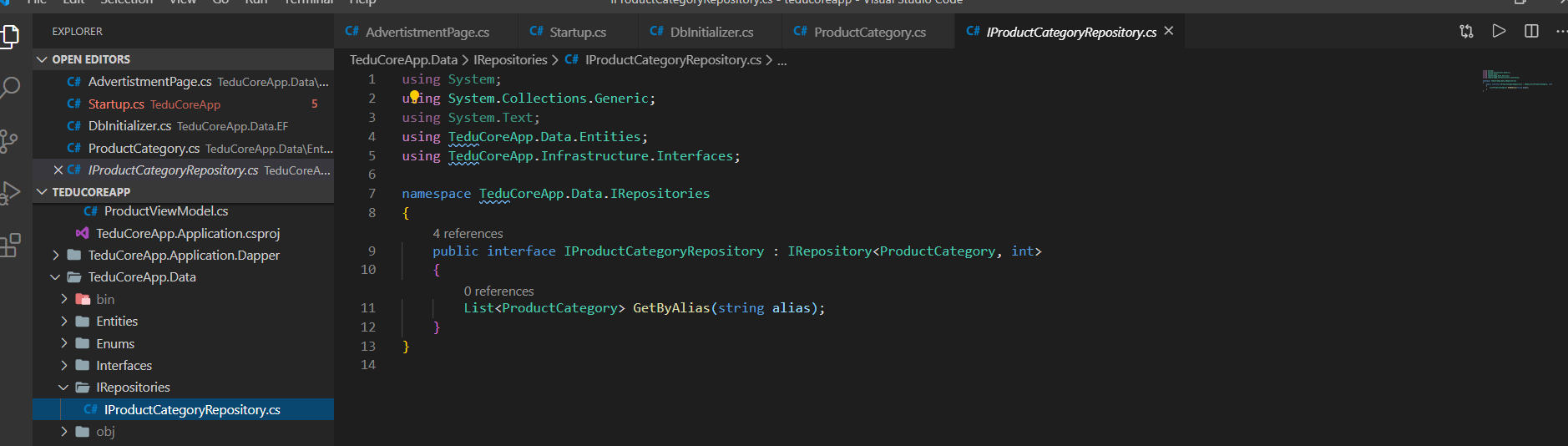
Trỏ tới ProductCategoryViewModel



Bỏ Column đi

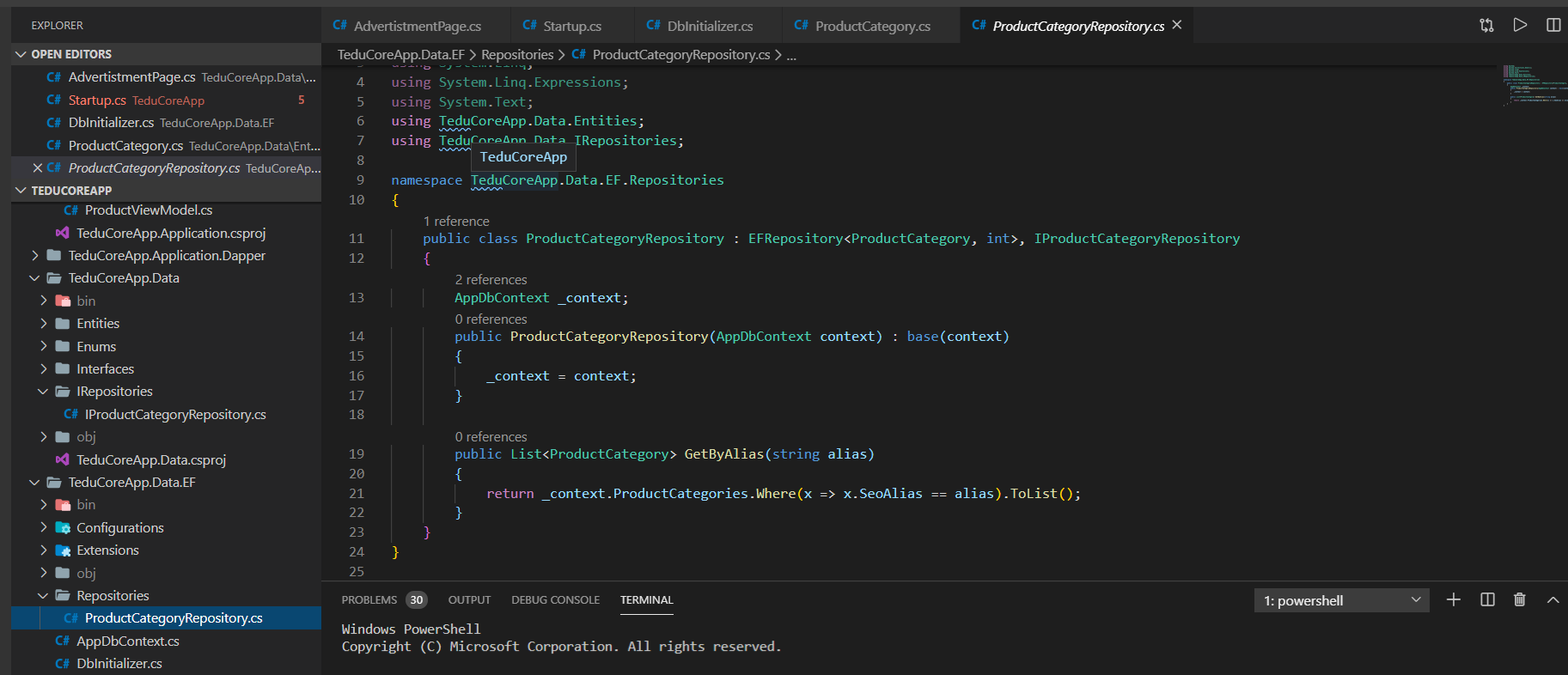
Vì là viewModel nên bỏ virtual, foreinKey, comlumn đi

Tên thuộc tính và validate giữ lại, tên thuộc tính giống thì mới map được



Triển khai IproductCategoryRepository triển khai từ Irepository (có các phương thức của cái này)

Thêm phương thức getbyalias

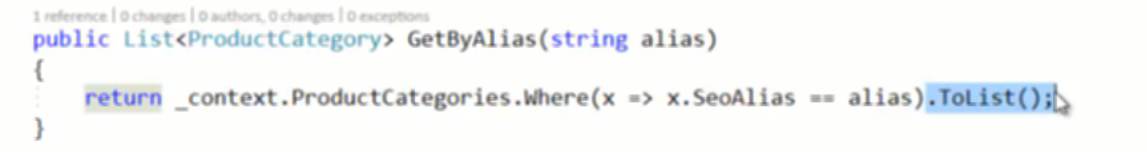


Triển khai ProductCategoryRepository

Kế thừa từ EFRepository nên có các method đã được triền khai trong class này, và kế thừa từ IproductCategoryRepository nên có thể implement các method đã thêm trong này

Vì EFRepository được implement từ IRepository mà IproductCategoryRepository là thừa kế từ IRepository, nên nhưng method trong EFRepository sẽ implement cho phần kế thừa trong IproductCategoryRepository -> chỉ cần implement những cái thêm vào IproductCategoryRepository

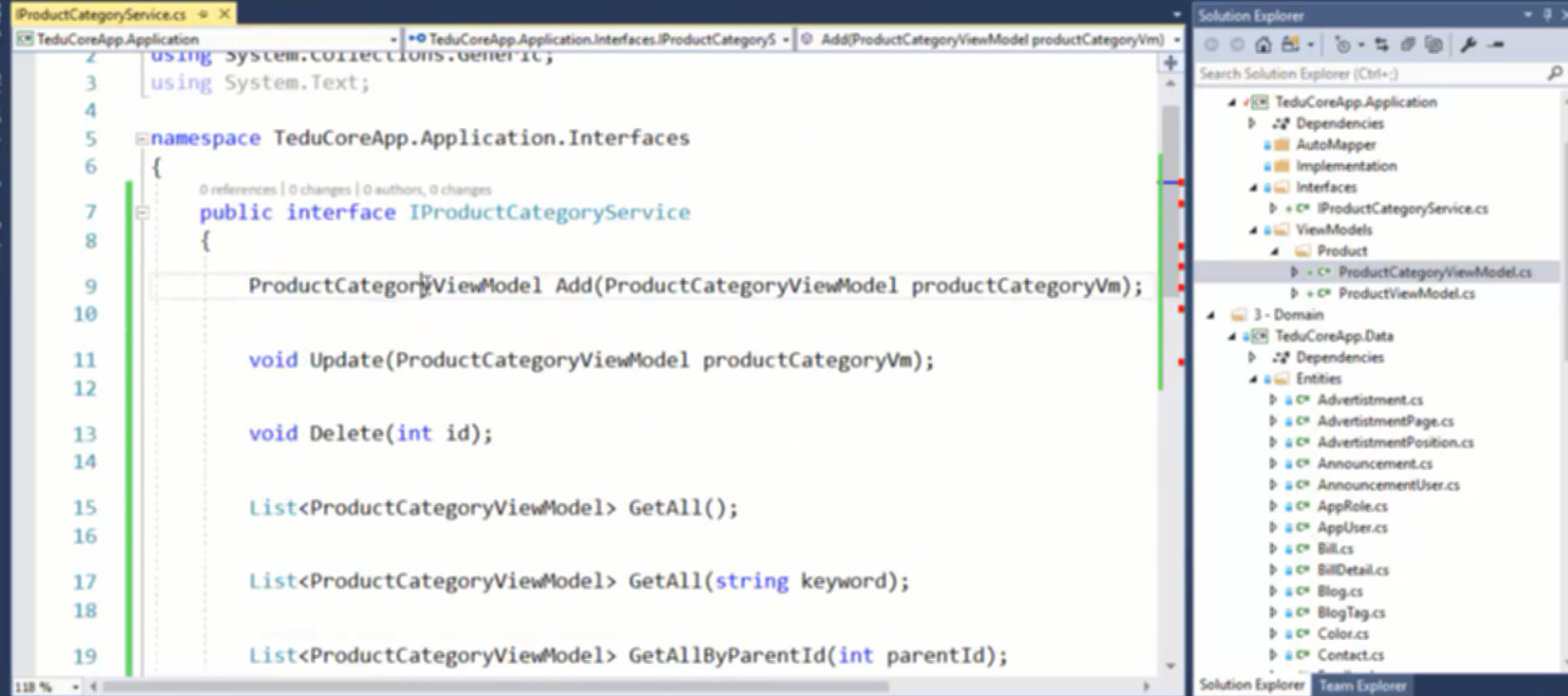
Sau này inject ProductCategoryRepository vào để xài trong app thông qua interface IproductCategoryRepository



Where chỉ mới áp điều kiện vào, Tolist, SingerDefault thì mới bắt đầu query

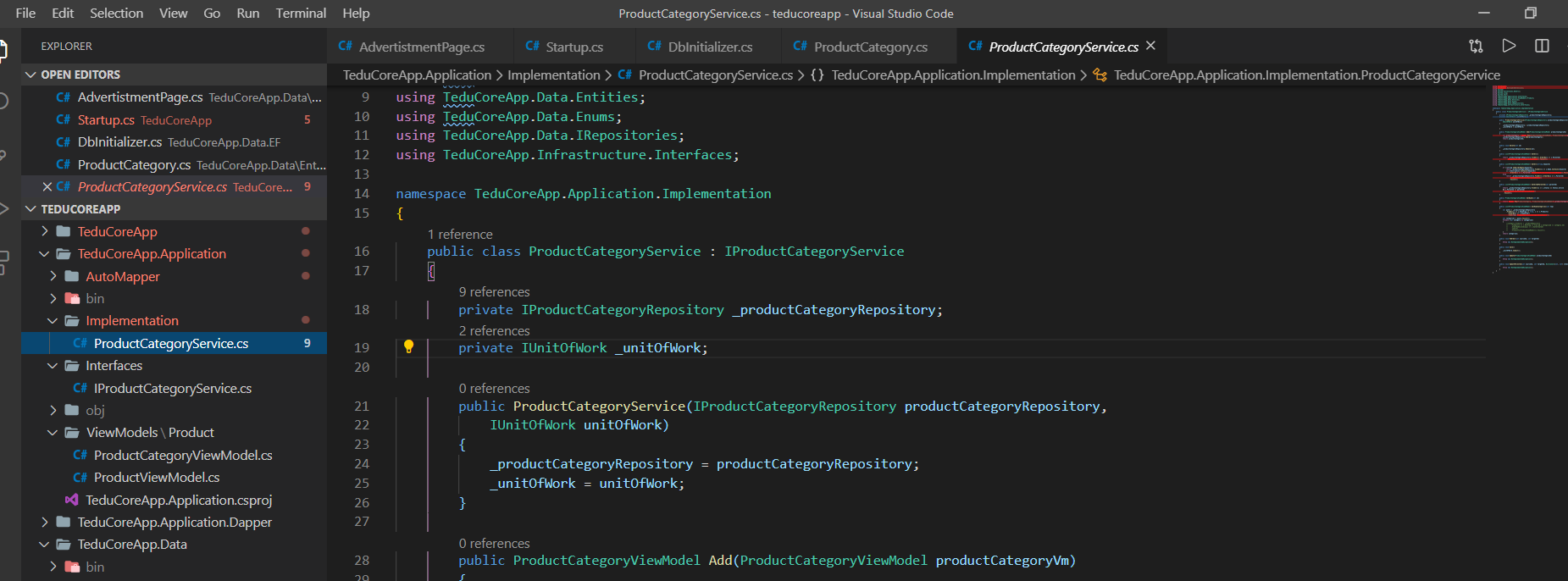
Mỗi model sẽ phải add vào app 1 repository và 1 service

viewModel là dữ liếu lấy ra từ db để hiển thị lên page thông qua controller



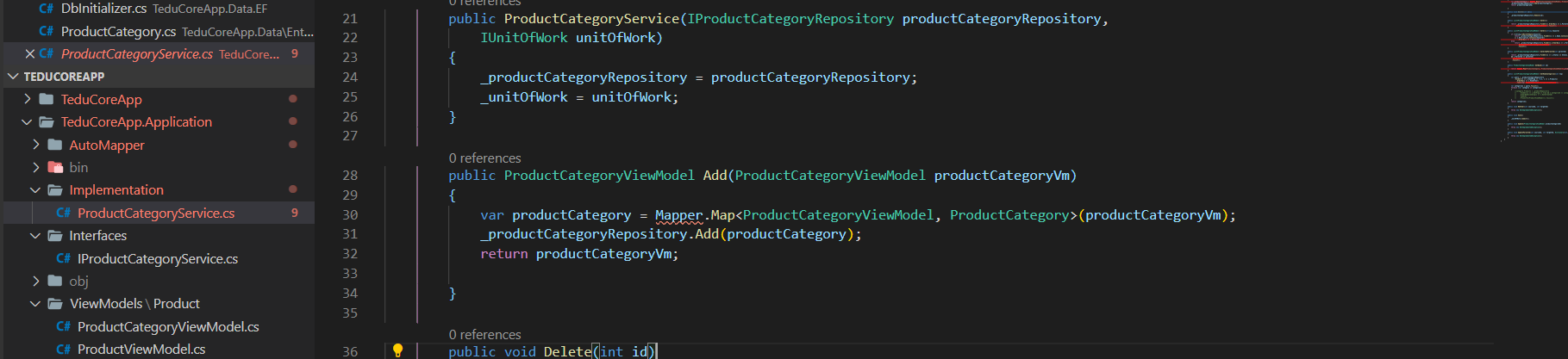
Tạo interface trong tầng Application, trong này sẽ định nghĩa các phương thức cho ProductCategory

Các phương thức trả hết về viewModal, controler sẽ gọi và dùng nhưng method trong này



Triển khai ProductCategory service

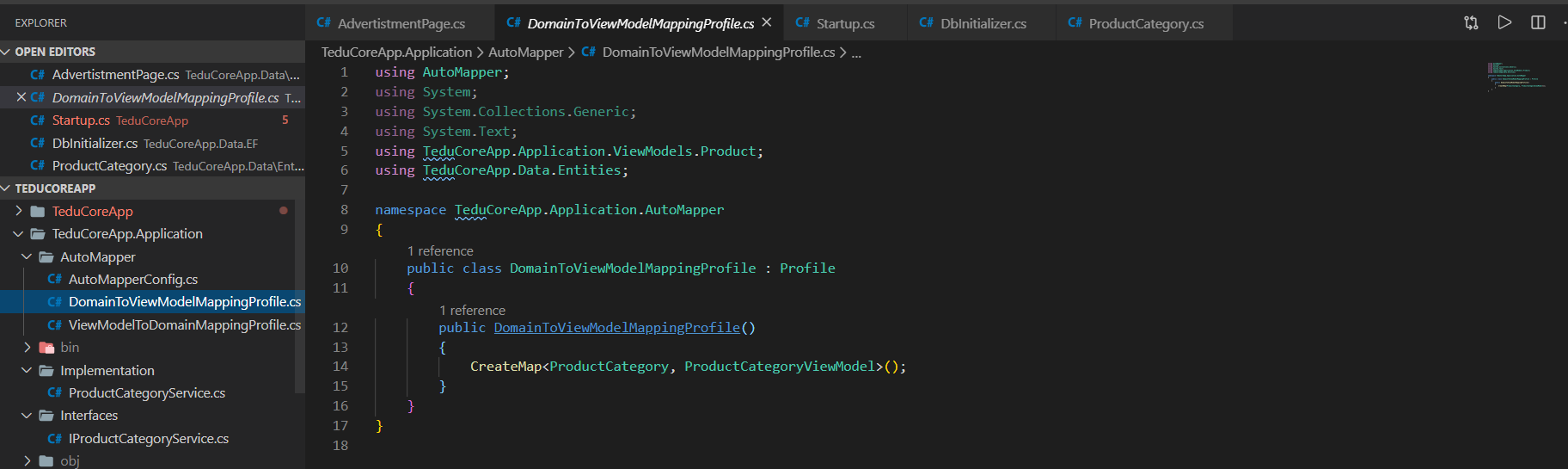
Tiêm vào IproductCategoryRepository và IUnitOfWork



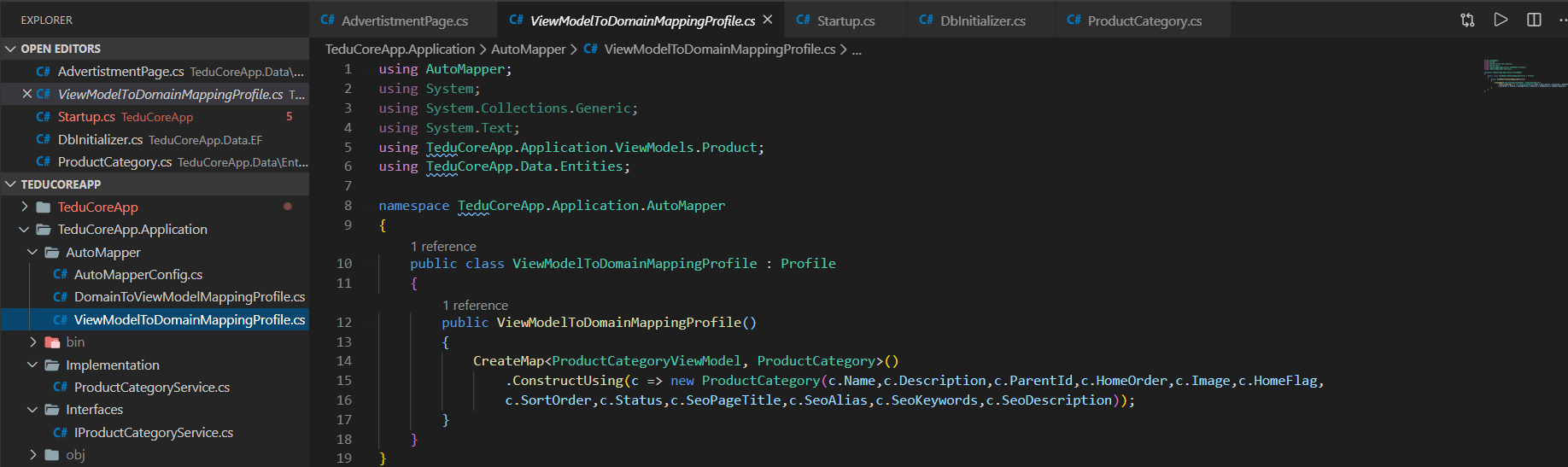
Lấy và thao tác từ viewModel -> cần phải map viewModel với entity

Cấu hình auto mapper

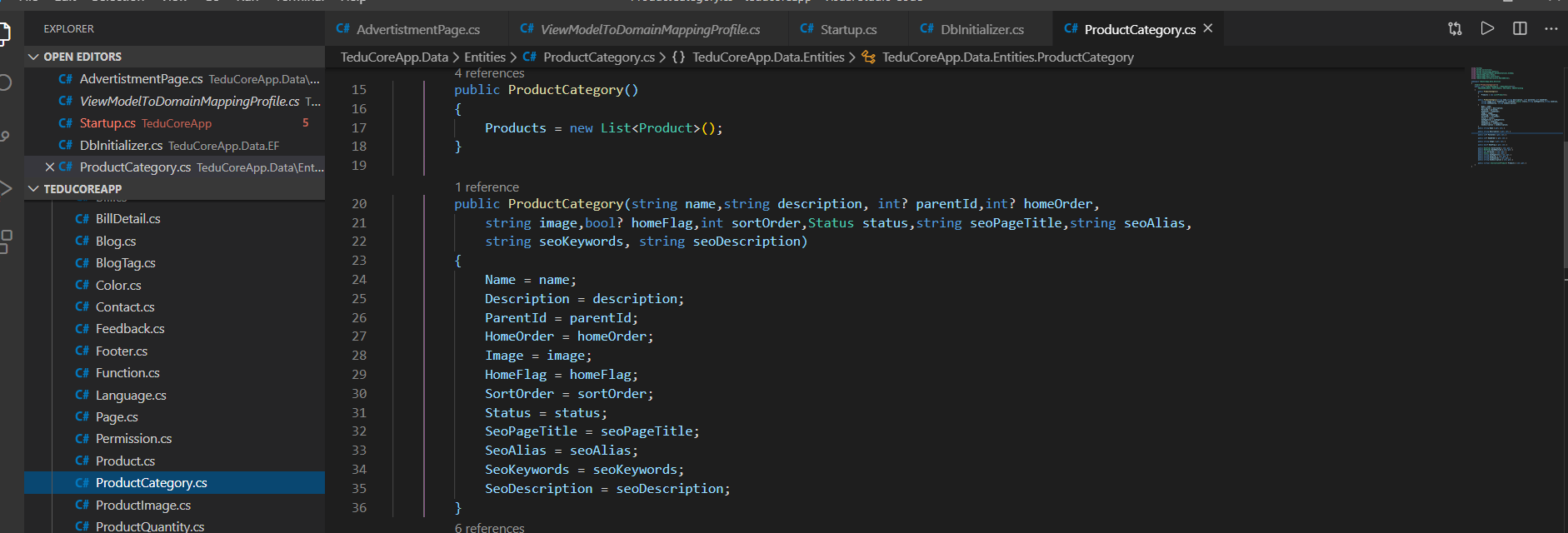
Cài AutoMapper nutget vào Application (6.1.1)



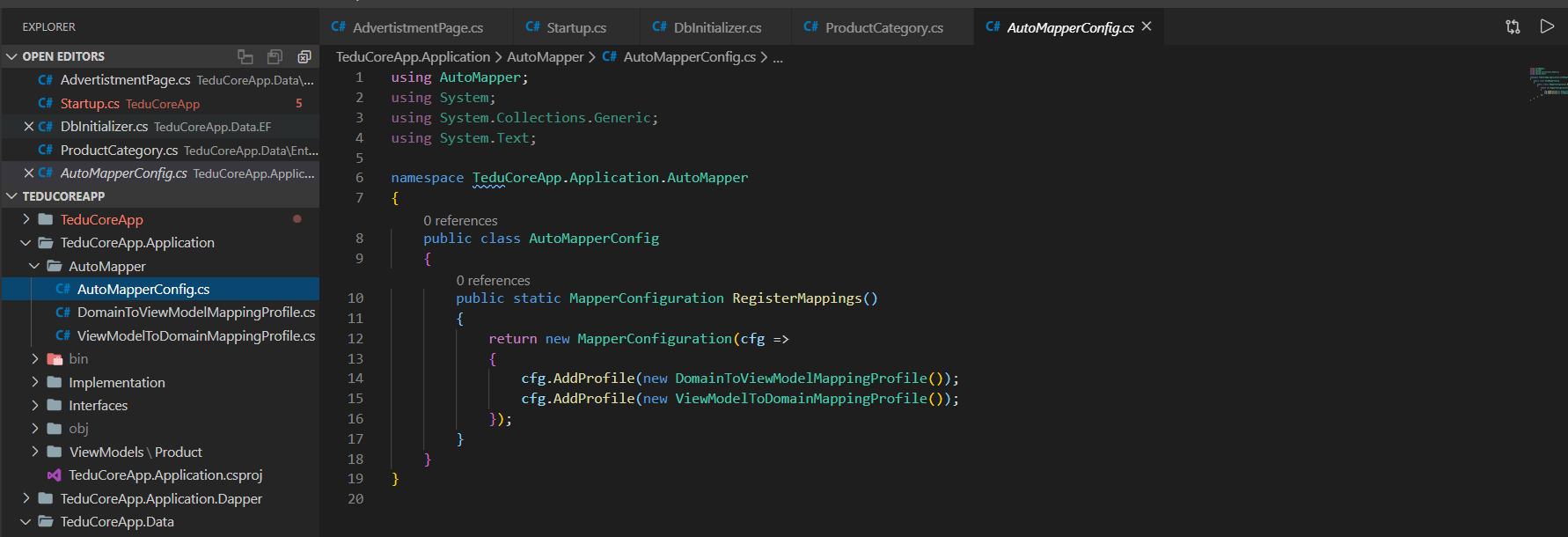
CreatMap của Profile



.ContruUsing để tạo ra contructor -> khi đưa c(viewModel) vào thì contructor sẽ tạo ra và map các giá trị của viewModel cho entity

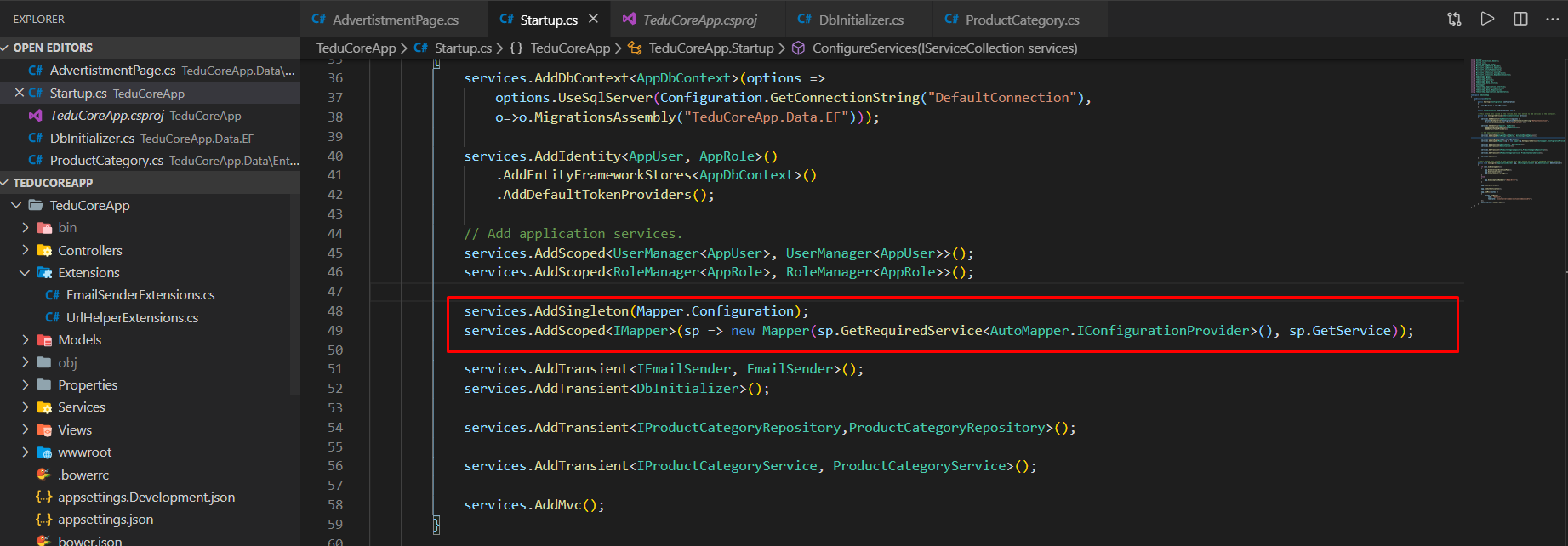


Viết thêm contructor và entity để ContruUsing có thể đưa giá trị vào



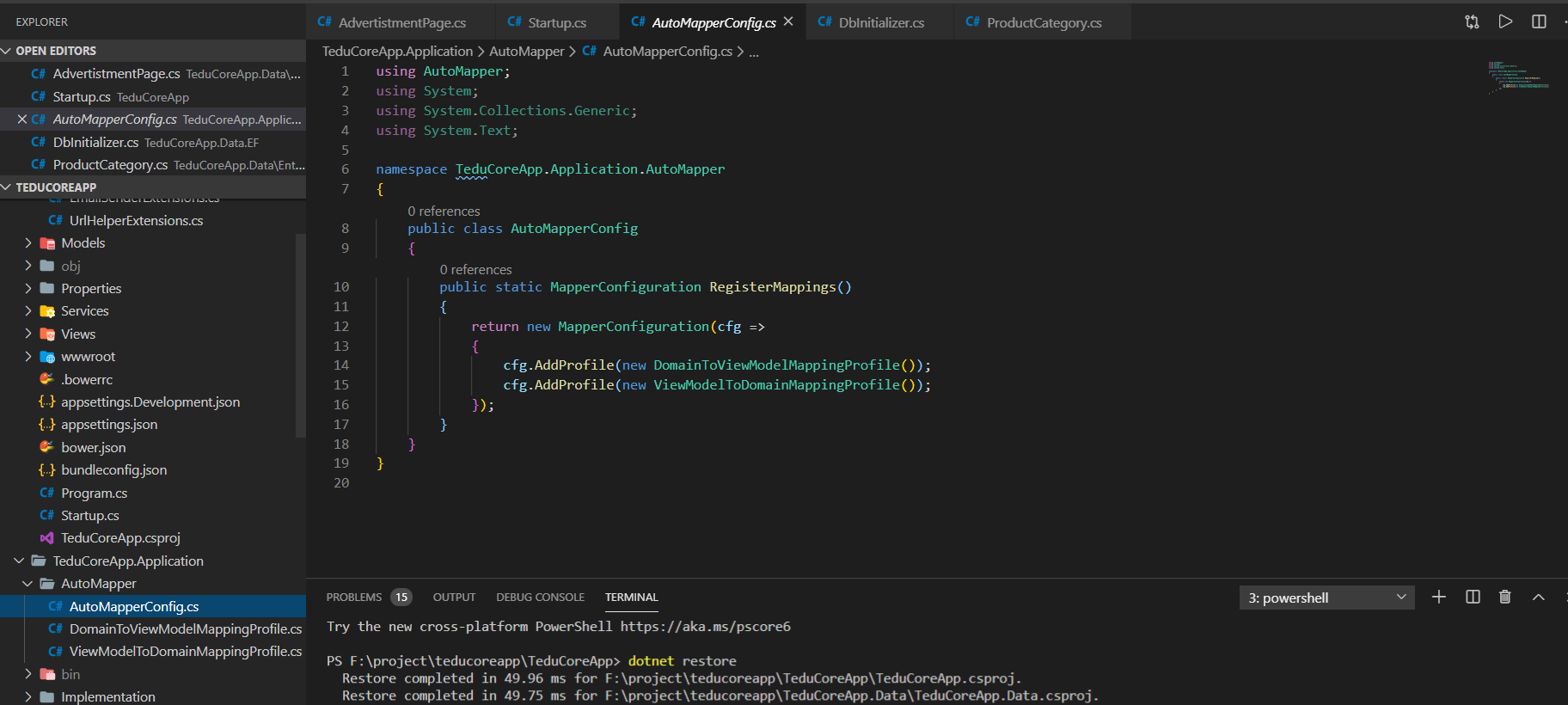
Khai báo AutoMapperConfig

Cài autoMapper vào App

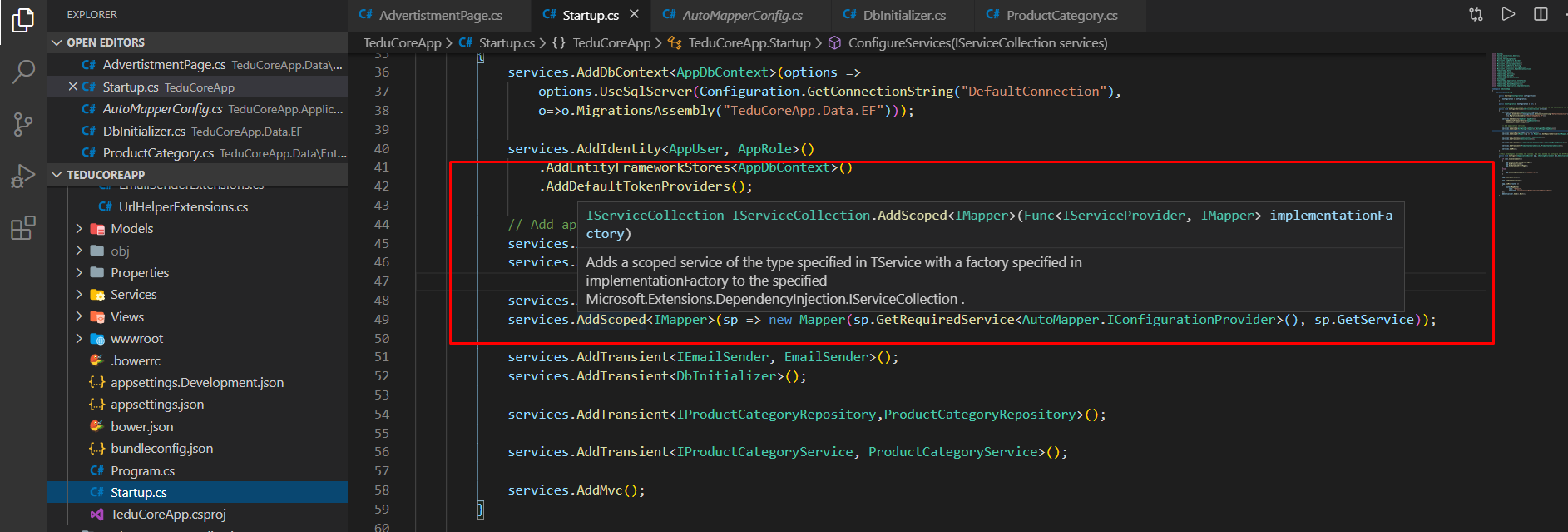


Add autoMapper vào

Thì instane Mapper.Configuration chính là cái này

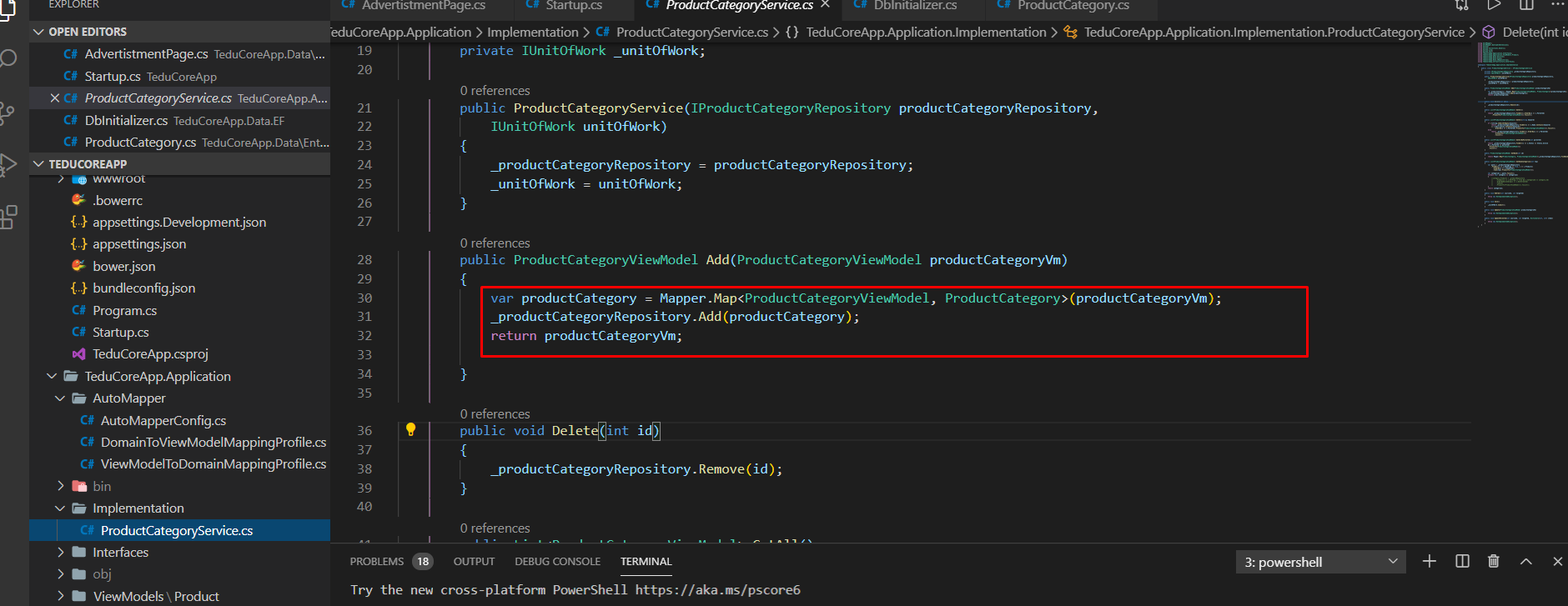


Vì phương thức này trả về configuration



ở đây cần truyền vào 2 tham số là IserviceProvider và Imapper -> dùng lambda truyền vào

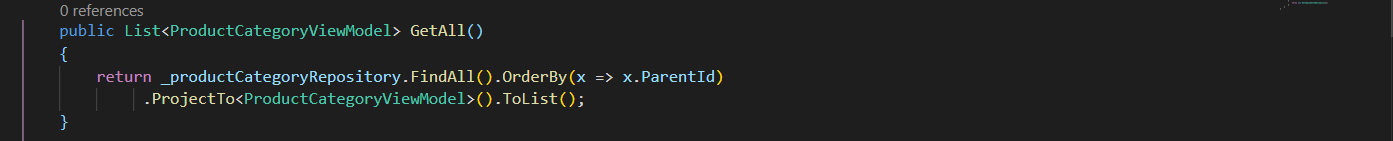
2 dòng này là cấu hình Imapper cho .net core



Map và sử dụng viewModel

Map từ viewModel sang entity với thông số truyền vào là 1 cái instance của viewModel

Tương tự với các phương thức khác



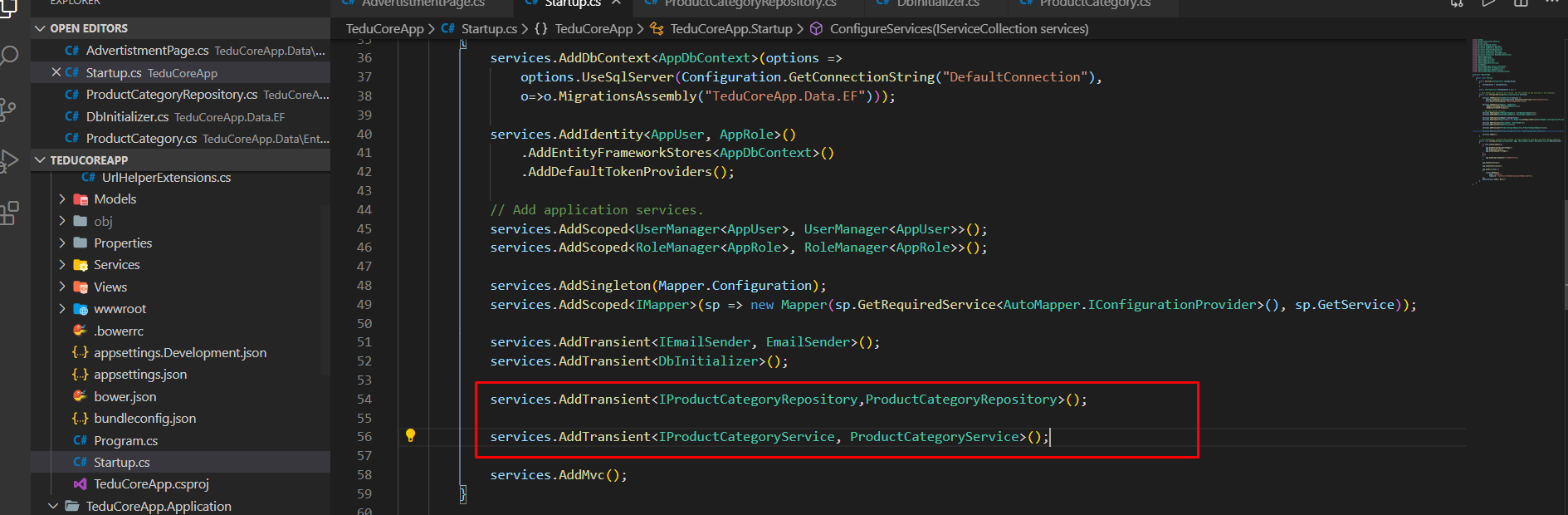
getAll lấy ra cả list cần phải dung ProjectTo viewModel

cái ProjectTo này chỉ dành cho IqueryAble

IqueryAble: là một đối tượng dùng để implement query



Lấy thêm product chỉ cần đẩy vào lambda



Khai báo service vào để sử dụng

Sau khi khai báo service xong thì có thể inject vào controller để xử lý được