Bahria University,

Karachi Campus



**LAB EXPERIMENT NO.**

**10**

**LIST OF TASKS**

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | Create a microservice for Products. |

**Task No. 1: Create a Microservice for Products.**

**Solution:**

**Product Controller:**

namespace ProductMicroservicee.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class ProductController : ControllerBase

{

private readonly IProductRepository \_productRepository;

public ProductController(IProductRepository productRepository)

{

\_productRepository = productRepository;

}

[HttpGet]

public IActionResult Get()

{

var products = \_productRepository.GetProducts(); return new OkObjectResult(products);

}

[HttpGet("{id}", Name = "Get")]

public IActionResult Get(int id)

{

var product = \_productRepository.GetProductByID(id); return new OkObjectResult(product);

}

[HttpPost]

public IActionResult Post([FromBody] Product product)

{

using (var scope = new TransactionScope())

{

\_productRepository.InsertProduct(product); scope.Complete();

return CreatedAtAction(nameof(Get), new { id = product.Id }, product);

}

}

[HttpPut]

public IActionResult Put([FromBody] Product product)

{

if (product != null)

{

using (var scope = new TransactionScope())

{

\_productRepository.UpdateProduct(product); scope.Complete();

return new OkResult();

}

}

return new NoContentResult();

}

[HttpDelete("{id}")]

public IActionResult Delete(int id)

{

\_productRepository.DeleteProduct(id); return new OkResult();

}

}

}

**Product Repository**:

namespace ProductMicroservicee.Repository

{

public class ProductRepository : IProductRepository

{

private readonly ProductContext \_dbContext;

public ProductRepository(ProductContext dbContext)

{

\_dbContext = dbContext;

}

public void DeleteProduct(int productId)

{

var product = \_dbContext.Products.Find(productId);

\_dbContext.Products.Remove(product); Save();

}

public Product GetProductByID(int productId)

{

return \_dbContext.Products.Find(productId);

}

public IEnumerable<Product> GetProducts()

{

return \_dbContext.Products.ToList();

}

public void InsertProduct(Product product)

{

\_dbContext.Add(product); Save();

}

public void Save()

{

\_dbContext.SaveChanges();

}

public void UpdateProduct(Product product)

{

\_dbContext.Entry(product).State = EntityState.Modified; Save();

}

}

}

**Connection String:**

{

"Logging": {

"LogLevel": {

"Default": "Information",

"Microsoft": "Warning",

"Microsoft.Hosting.Lifetime": "Information"

}

},

"AllowedHosts": "\*",

"ConnectionStrings": {

"ProductDB": "Data Source=Muaz-Shahzad\\SQLEXPRESS;Initial Catalog=ProductsDB;Integrated Security=True;"

}

}

**Output:**

**A screenshot of a computer

Description automatically generated with medium confidence**