

T item = new T();

T Factory.Resolve<T>()

Type	Value
IProductRepository	DbProductRepository
ICustomerRepository	FakeCustomerRepository

Controller

View()

ApiController

Ok()

NotFound()

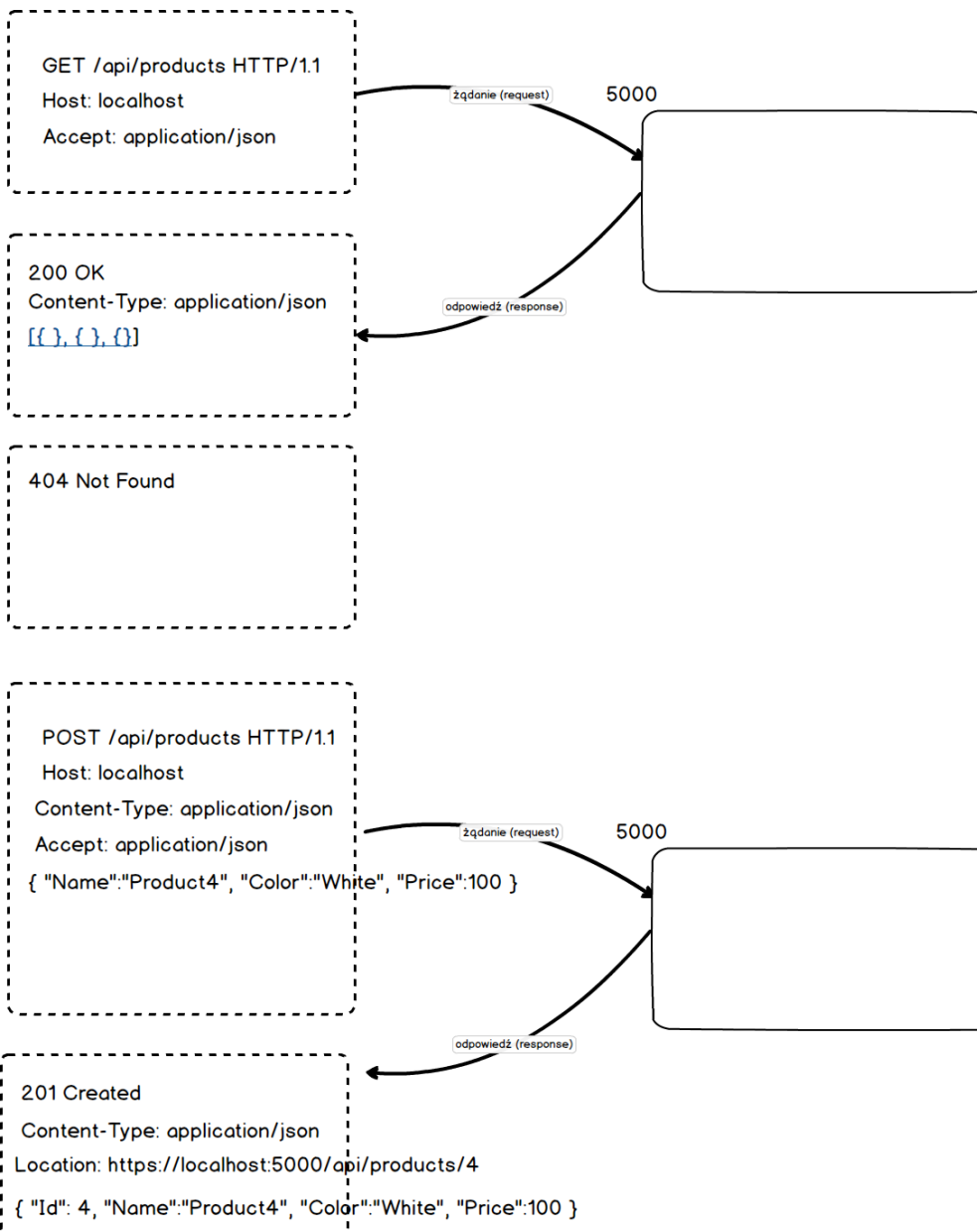
ControllerBase

Ok()

NotFound()

Controller : ControllerBase

View()



Akcje HTTP (HTTP Methods)

GET - pobierz

POST - utwórz

PUT - zamień

PATCH - zmień

DELETE - usuń

HEAD - czy istnieje

myblog.com/posts/net-core

myblog.com/posts/net-core/hello-world

myblog.com/posts/net-core/web-api

myblog.com/posts/ef-core

SOAP, WCF	RPC (Remote Procedure Calls)
REST API	Resources

POST `https://localhost:5000/api/reports`

GET `https://localhost:5000/api/reports/{reportId}`

GET `api/customers/{customerId}/products`

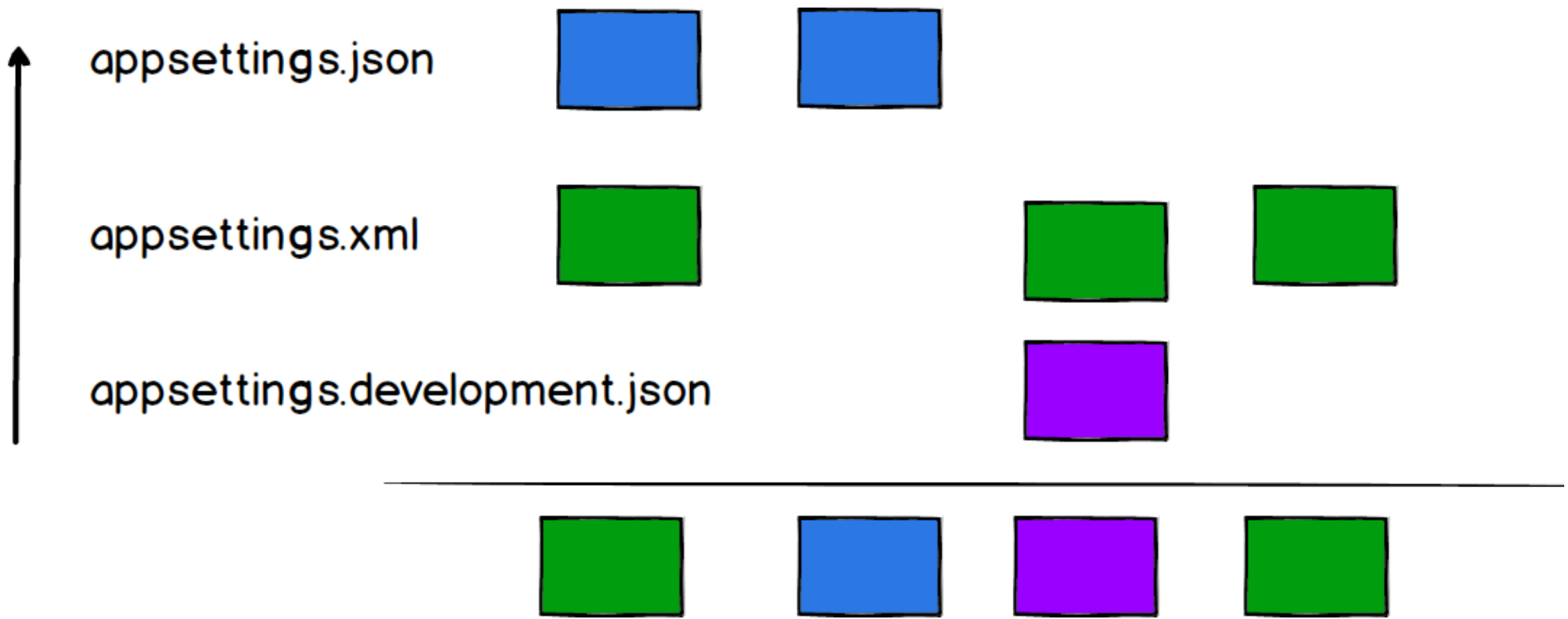
Zadanie #1

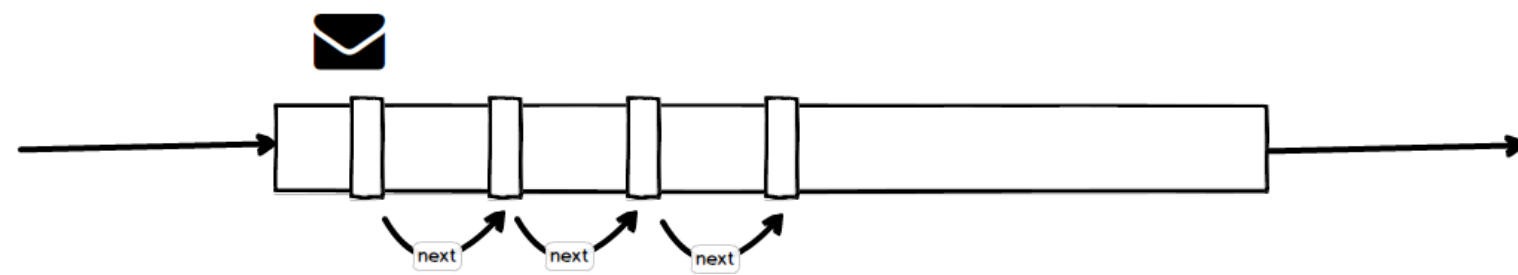
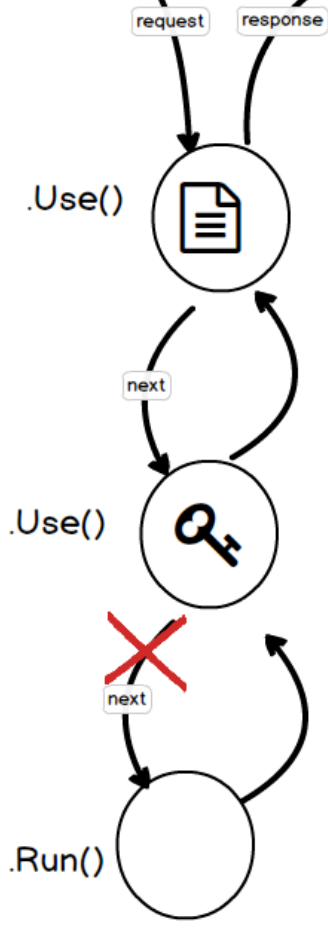
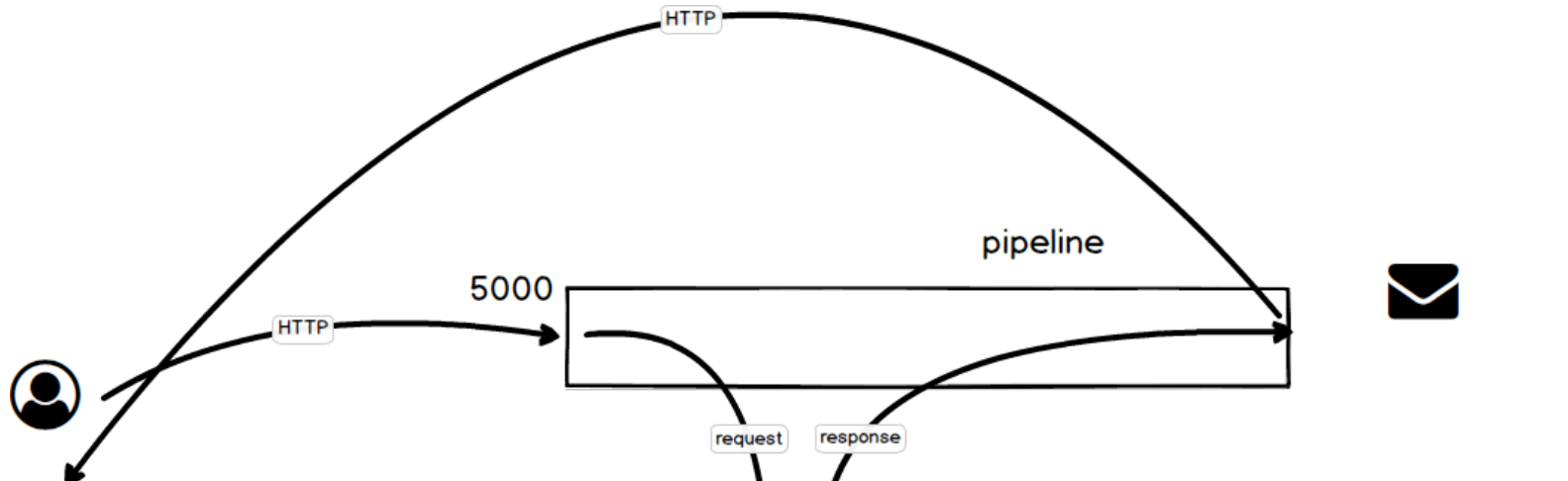
- 1 Dodawanie klienta
- 2 Pobieranie listy klientów
- 3 Pobieranie klienta na podstawie id
- 4 Modyfikacja klienta
- 5 Usuwanie klienta na podstawie id

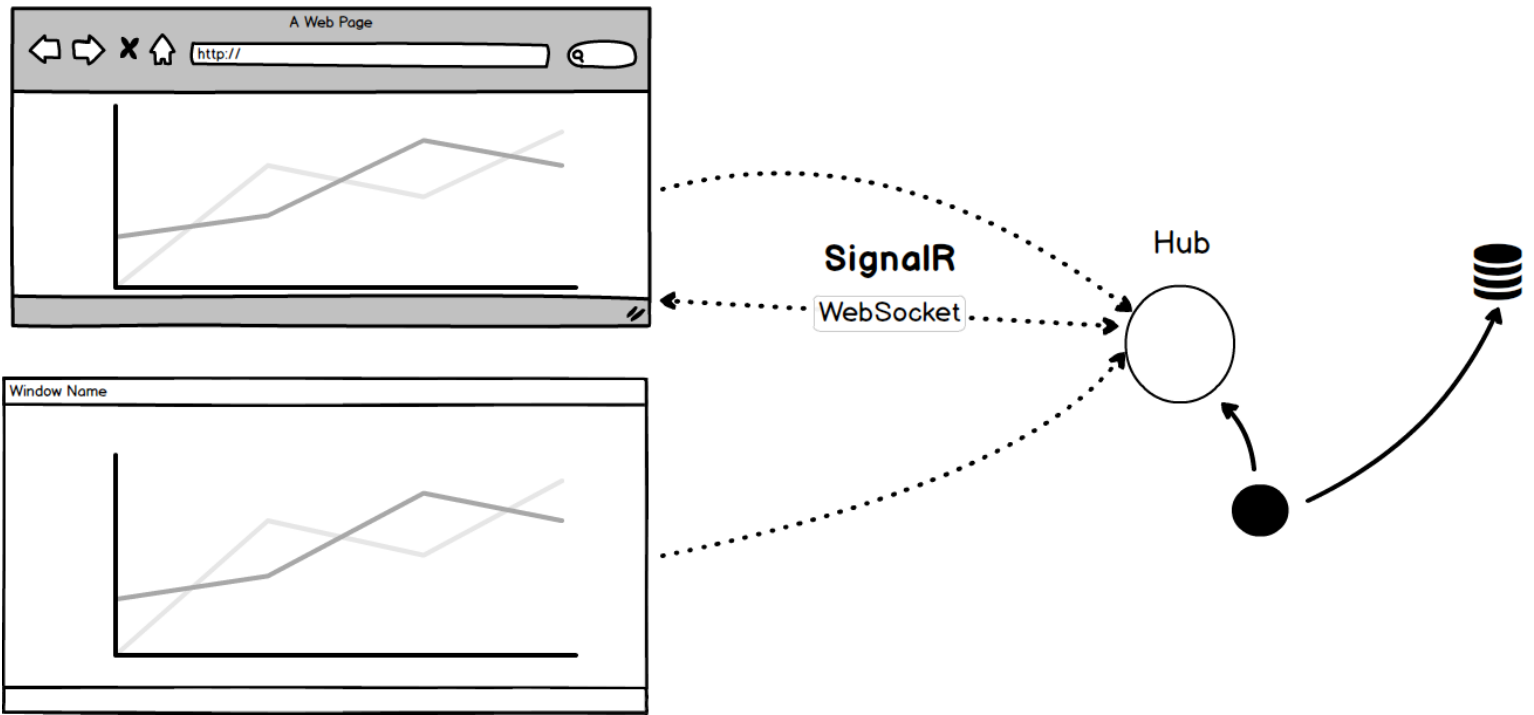
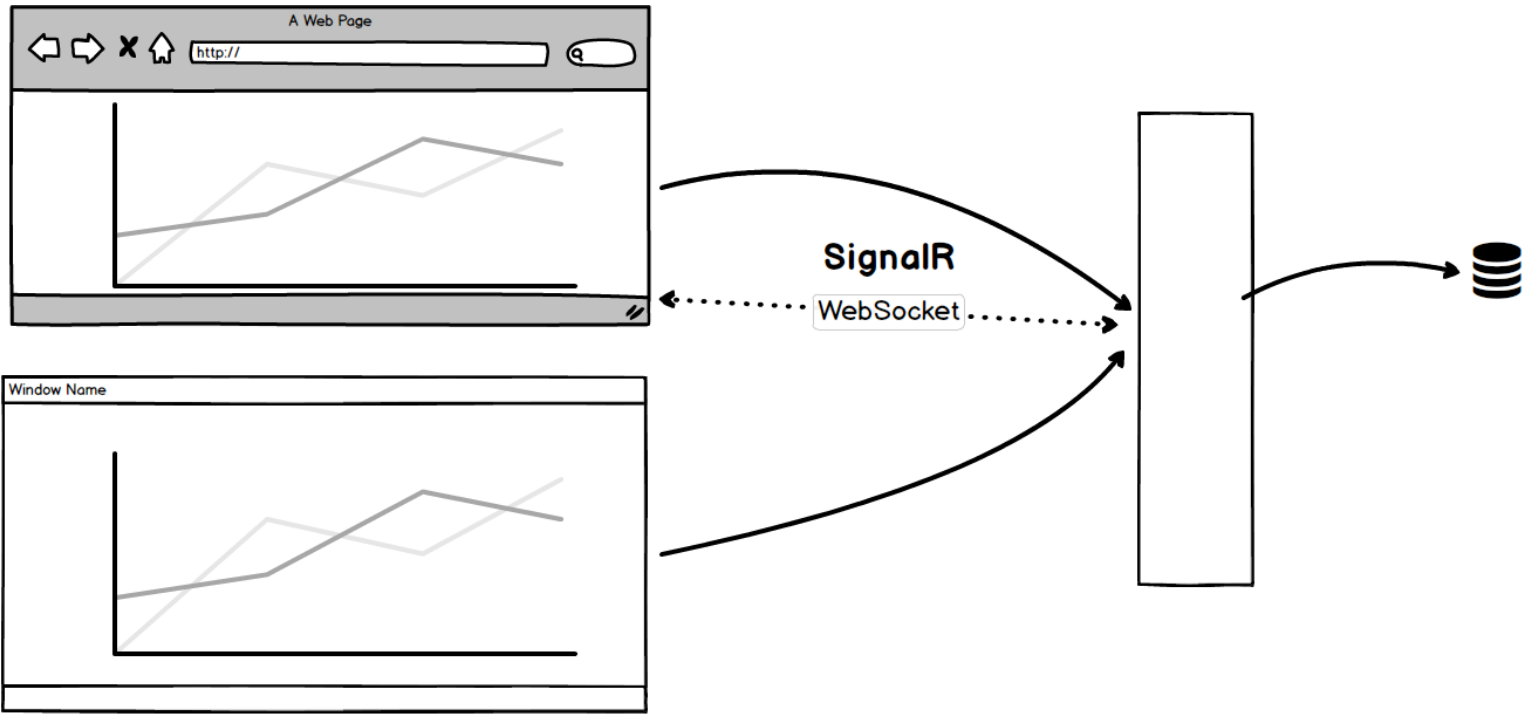
Domain

Customer
Id
FirstName
LastName
Email
Gender
IsRemoved

Gender = { Male, Female }

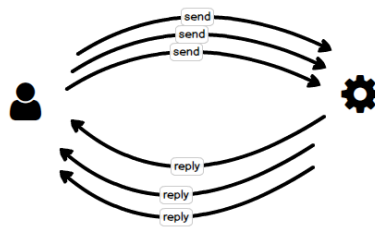
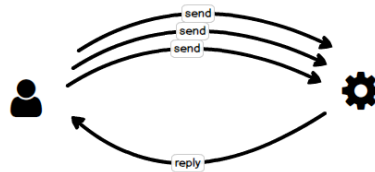
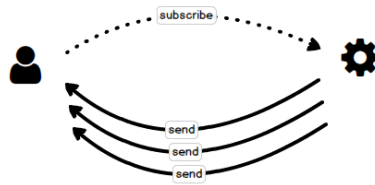
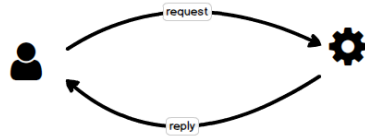
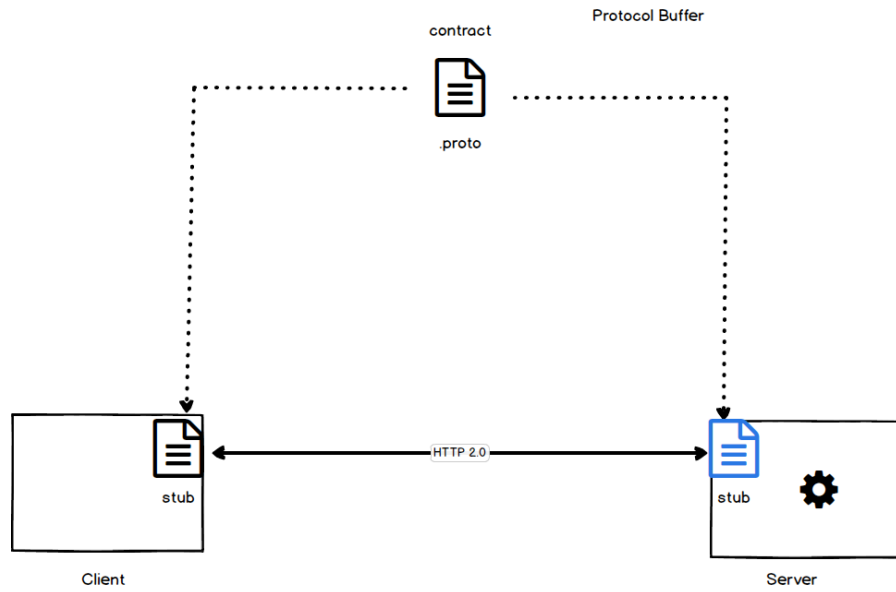






gRPC

(Google Remote Procedure Calls)

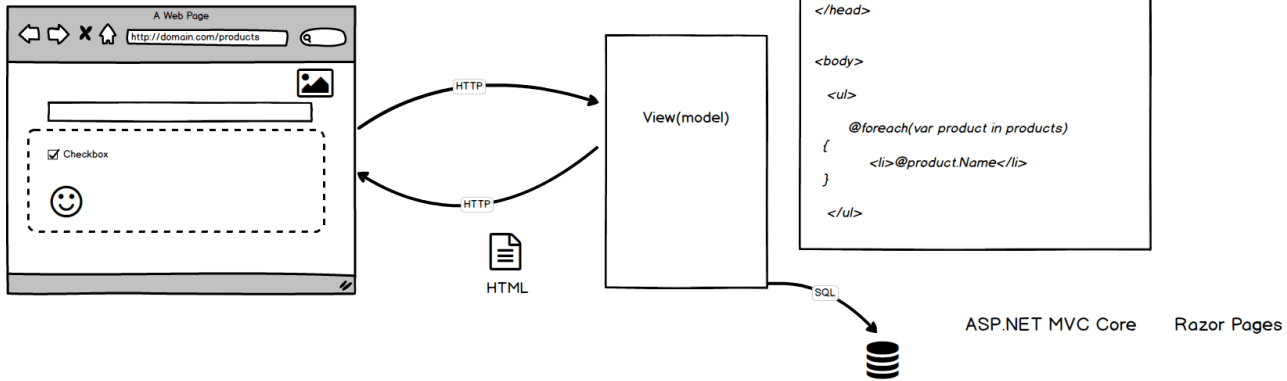


Code First



interface

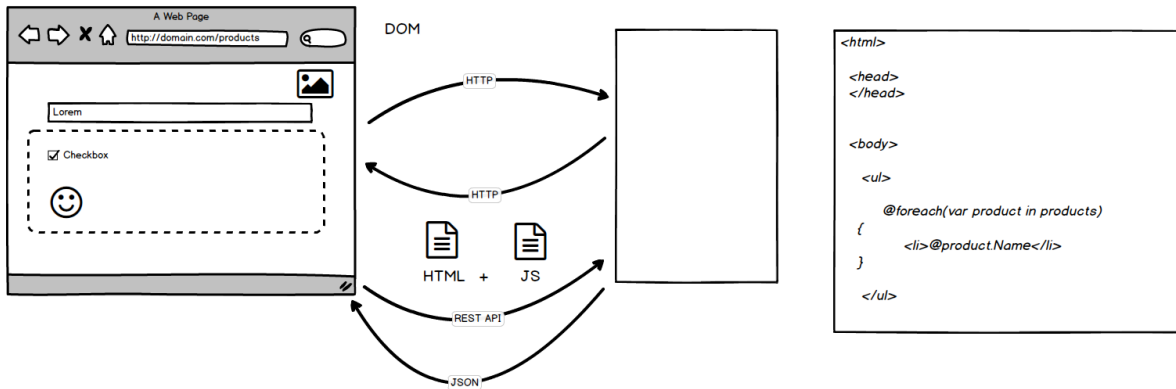
Template (Razor Syntax)



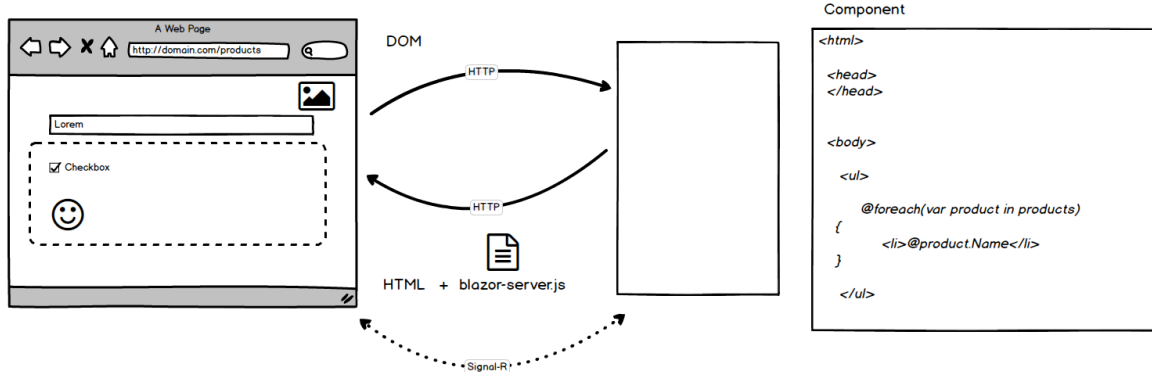
JS (Angular, React, Vue)

TypeScript

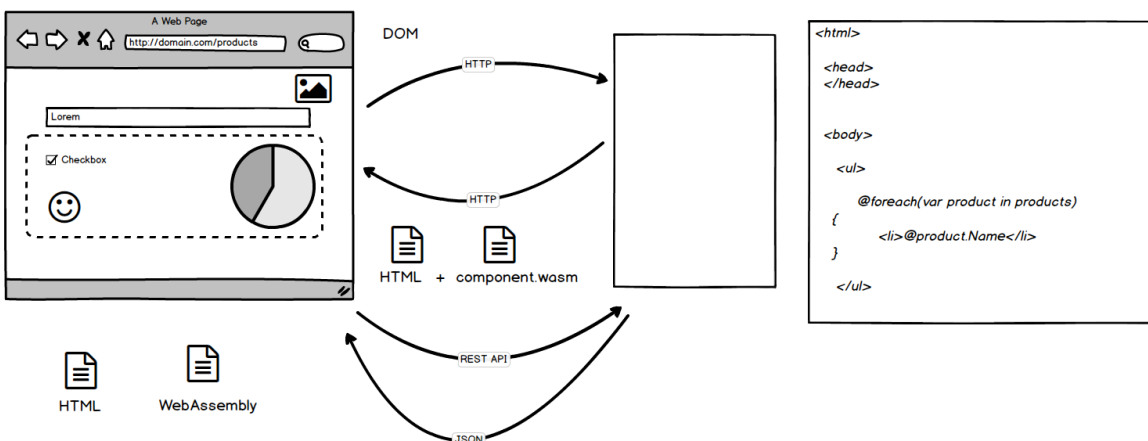
JavaScript



Blazor (Server-Side)



Blazor (Client-Side) Web Assembly

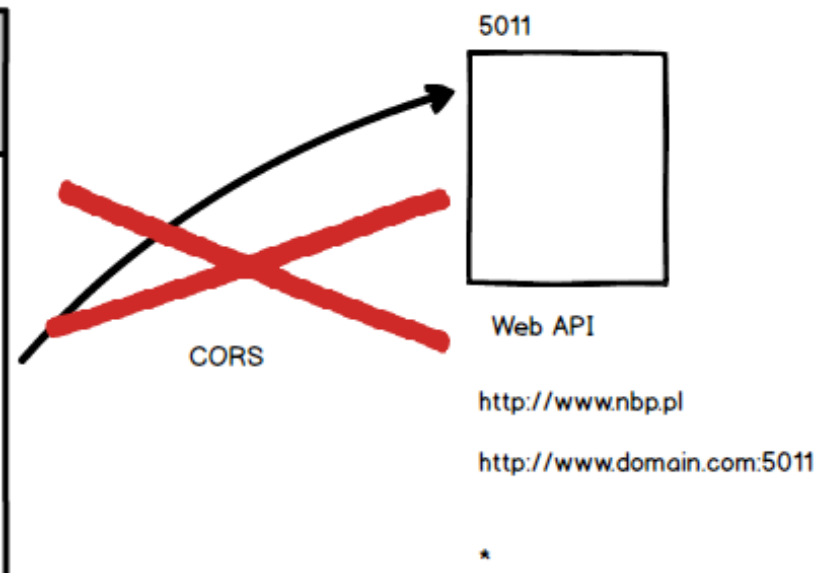
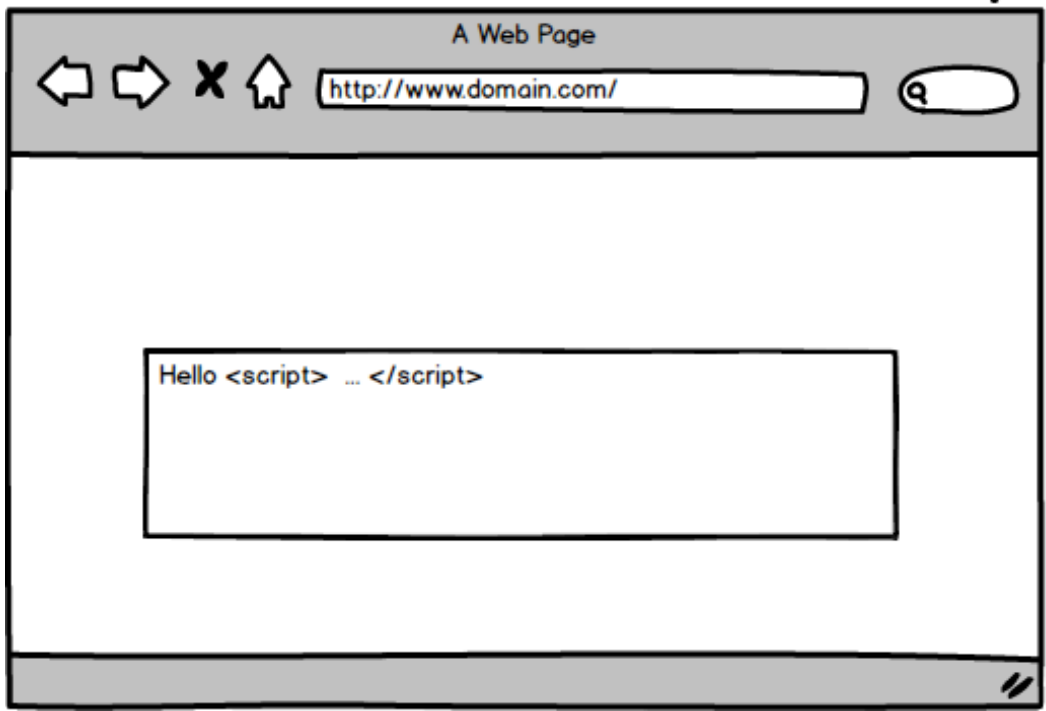
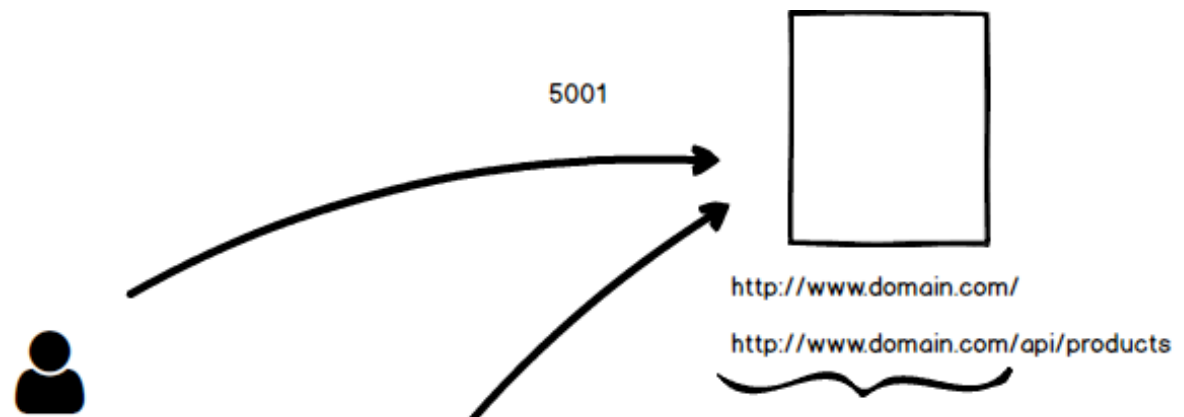


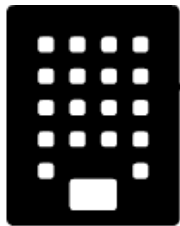


Title



Body





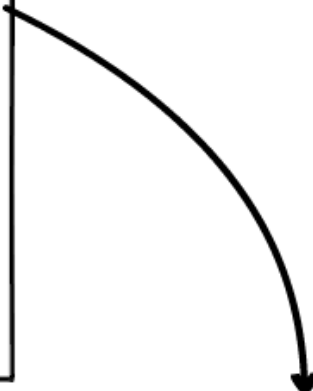
John Smith

ClaimType	Value
kat	B
kat	C
termin-waznosci	
Role	developer
Role	trainer

IPrincipal

IIdentity Identity { get; }

bool IsInRole(string)



IIdentity

string Name { get; }

bool IsAuthenticated { get; }

string AuthenticationType { get; }

POST api/token

json

{ login, hasło }

form

login: ;hasło:



53f83n903890385034535

GET api/customers

Authorize: Bearer {token}



53f83n903890385034535

POST api/token

json { login, hasło }

form login: ;hasło:

GET api/products

Authorization: Bearer {token}

IdentityDbContext<IdentityUser, IdentityRole>



UserManager<IdentityUser>

