**Vs code xtensions :**

C# for visual studio

**Reformatting :**

1. On Windows Shift + Alt + F.

**Vscode extensions:**

C# exteisons - by josh kreativ

auto rename tag – josh hun

bracket pair colorizer -conrasds

material icon theme -phiillip keff

nuget gallery - pcilo

sqlite – alex valzz

postgres sql

API:

**commands:**

dotnet run

dotnet watch run

dotnet tool list -g

**entity framework:**

dotnet tool update --global dotnet-ef --version 6.0.1

dotnet-ef migrations add InitialCreate -o Data/Migrations

dotnet-ef database update

dotnet ef migrations add fixproductDescriptionTypo -context storecontext

dotnet ef database drop

dotnet ef migrations add basketEntityAdded

dotnet ef migrations remove

dotnet ef migrations add OrderEntityAdded -o Data/Migrations

<https://docs.microsoft.com/en-us/ef/core/managing-schemas/migrations/managing?tabs=dotnet-core-cli>

asp core middleware :

<https://docs.microsoft.com/en-us/aspnet/core/fundamentals/middleware/?view=aspnetcore-6.0>

**nugets:**

microsoft.entityframework.sqlite

microsoft.entityframework.core.design

**references:**

<https://www.nuget.org/packages/dotnet-ef/>

dotnet tool install --global dotnet-ef --version 6.0.1

API:

Table

Description automatically generated

REACT:

webSites:

Reactjs.org

<http://json2ts.com/>

<https://reactjs.org/docs/faq-structure.html> - folder structure -group by features

material ui:

<https://mui.com/> - material UI library

<https://mui.com/getting-started/installation/>

<https://mui.com/customization/typography/>

npm install @mui/material @emotion/react @emotion/styled

npm install @mui/lab

react tostify:

<https://www.npmjs.com/package/react-toastify>

npm install react-toastify

Commands:

New react proj command:

npx create-react-app client –template typescript

run :

cd client - > npm start

installing routing :

npm install [react-router-dom@5.3.0](mailto:react-router-dom@5.3.0)

reactrouter.com API

<https://reactrouter.com/docs/en/v6/api>

reactrouter tester:

<https://v5.reactrouter.com/web/example/auth-workflow>

npm i react-router-dom@5.3.0 @types/react-router-dom@5.3.2

REDUX:

<https://redux.js.org/>

npm install redux react-redux

notes:\

review react tooklit(search for tutorial there)

Graphical user interface, diagram, application

Description automatically generated

REDUX :

<https://redux.js.org/style-guide/style-guide/>

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

Diagram

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

REDUX:

<https://redux.js.org/>

npm install redux react-redux

redux toolkit:

<https://redux.js.org/redux-toolkit/overview>

npm install @reduxjs/toolkit

redux thunk:

[GitHub - reduxjs/redux-thunk: Thunk middleware for Redux](https://github.com/reduxjs/redux-thunk)

npm install redux-thunk

chrome extensions :

react devtools

redux devtools

Graphical user interface, text, application, email

Description automatically generated

Diagram

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated with medium confidence

Text

Description automatically generated

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

TOKEN DECIPHER SITE :

<https://jwt.io/>

Diagram

Description automatically generated with low confidence

Graphical user interface, text, application

Description automatically generated

{

"username": "bob",

"password": "Pa$$w0rd"

}

Regular expression site:

<https://regexlib.com/?AspxAutoDetectCookieSupport=1>

yup validations :

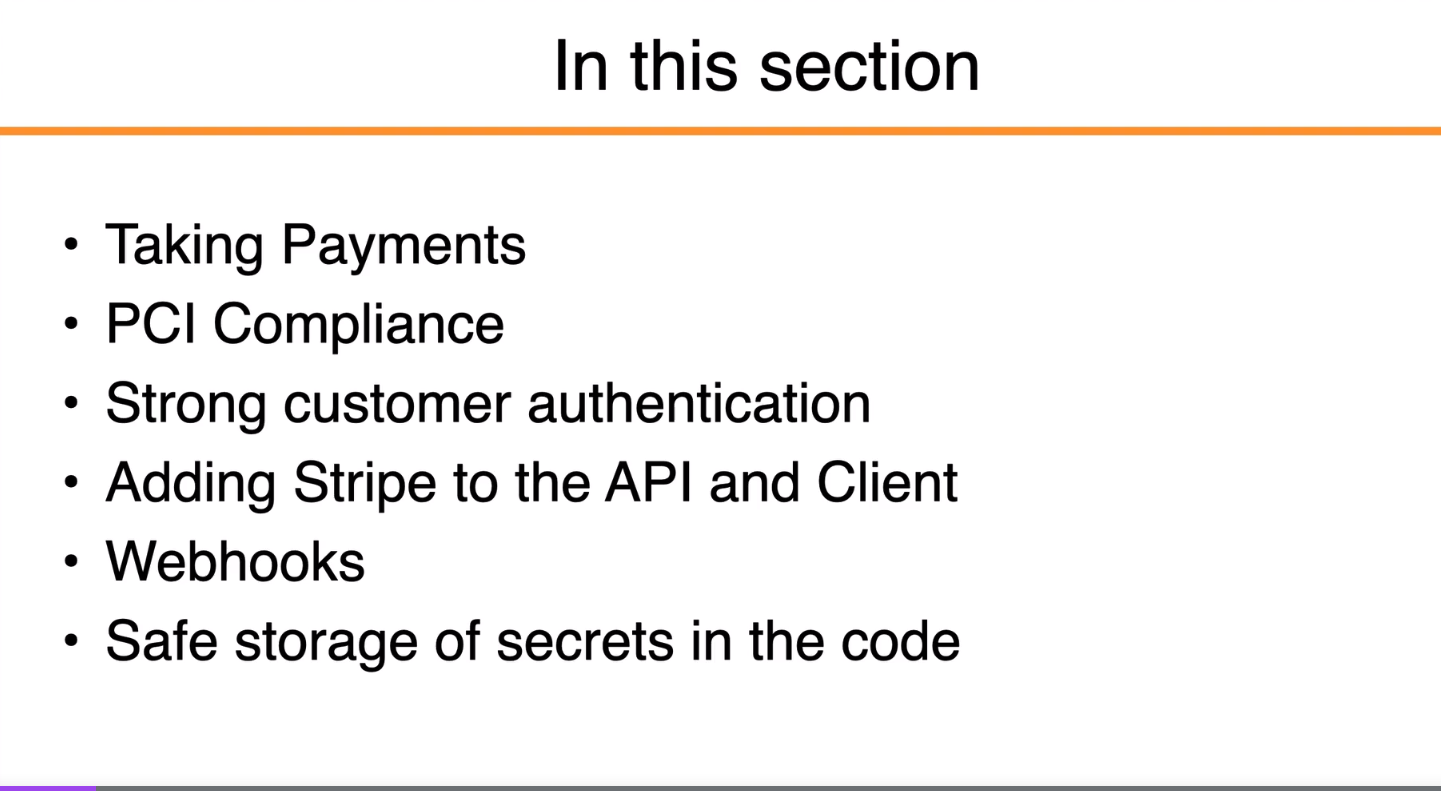
npm install @hookform/resolvers yup

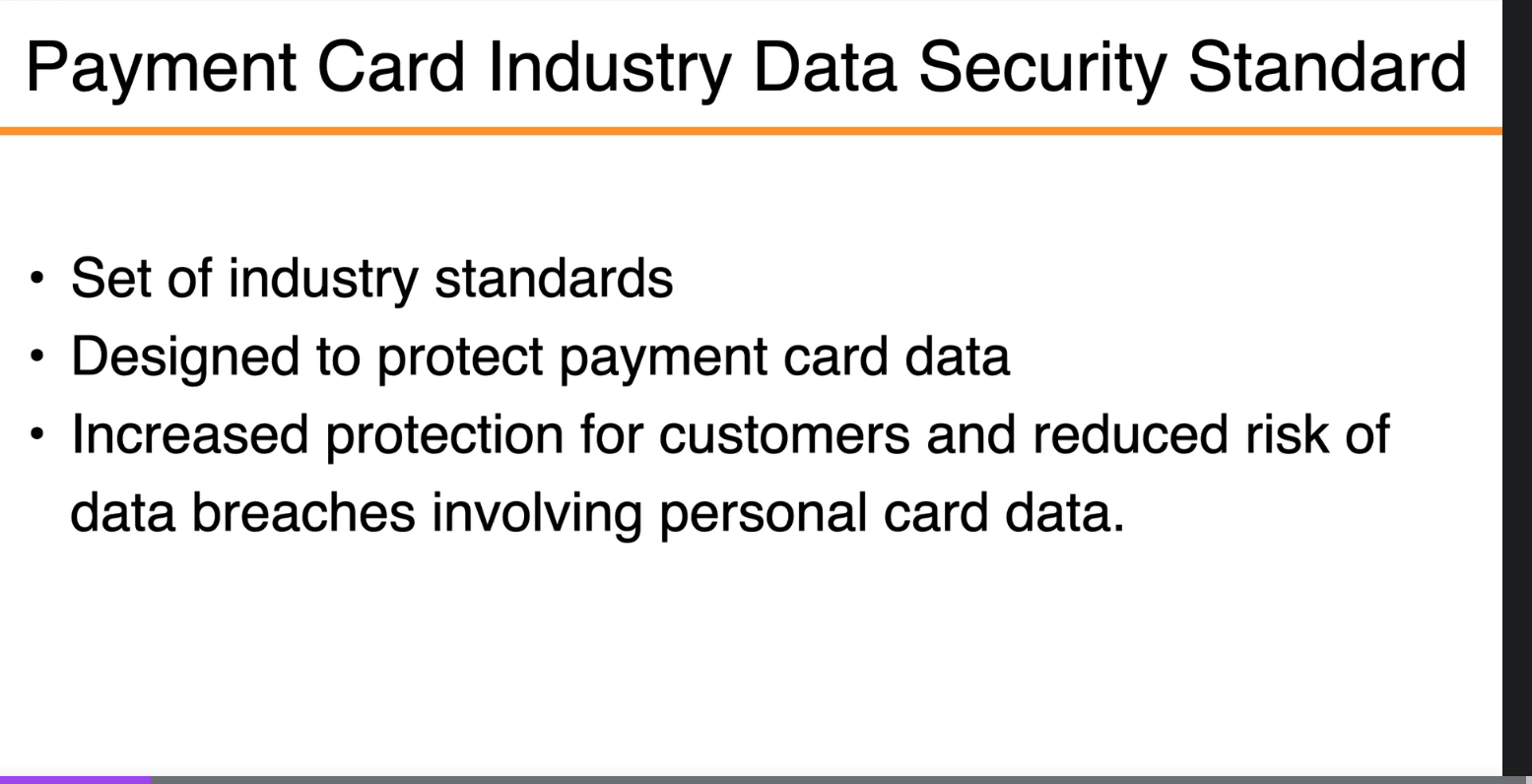
React hook form :

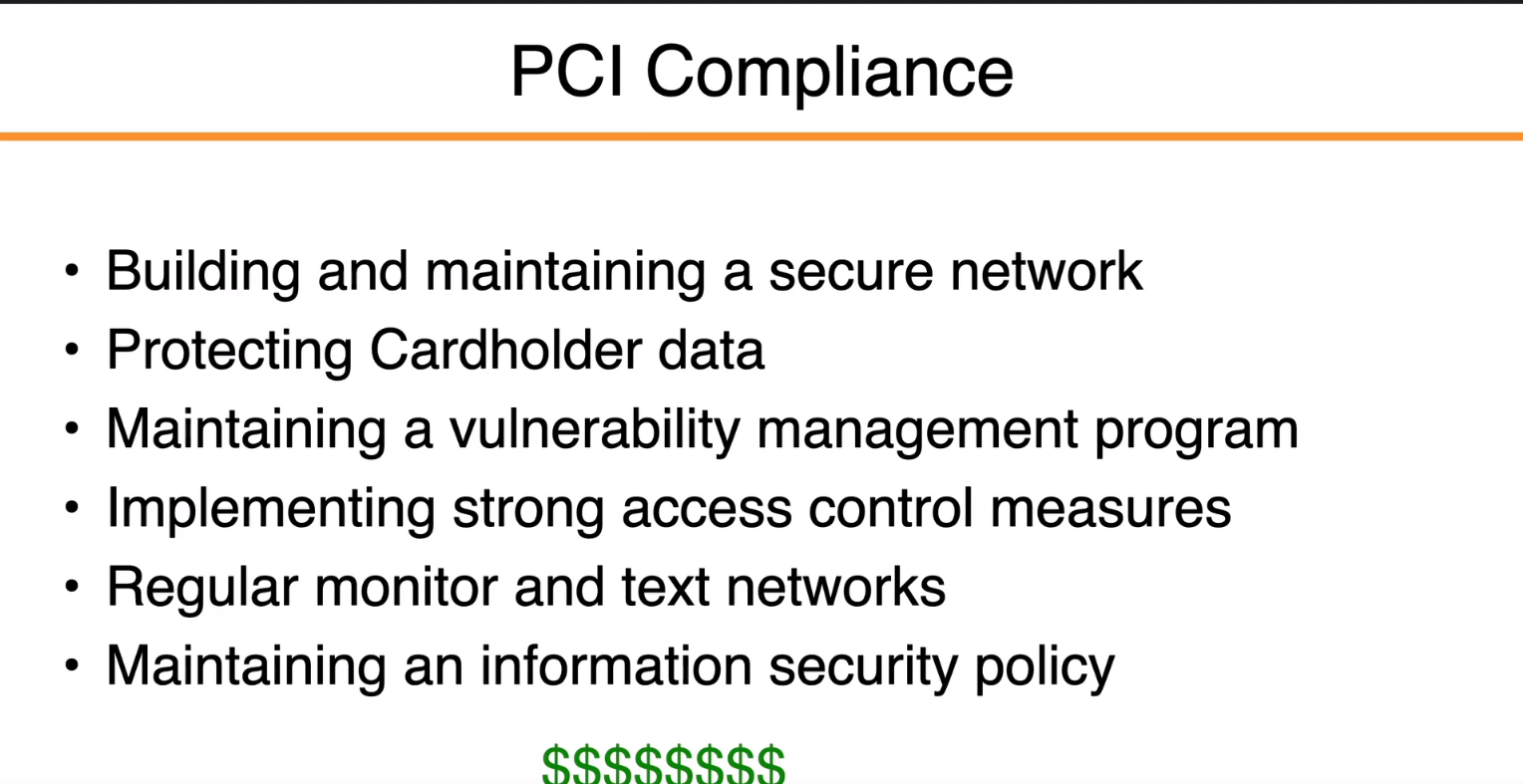
<https://react-hook-form.com/>

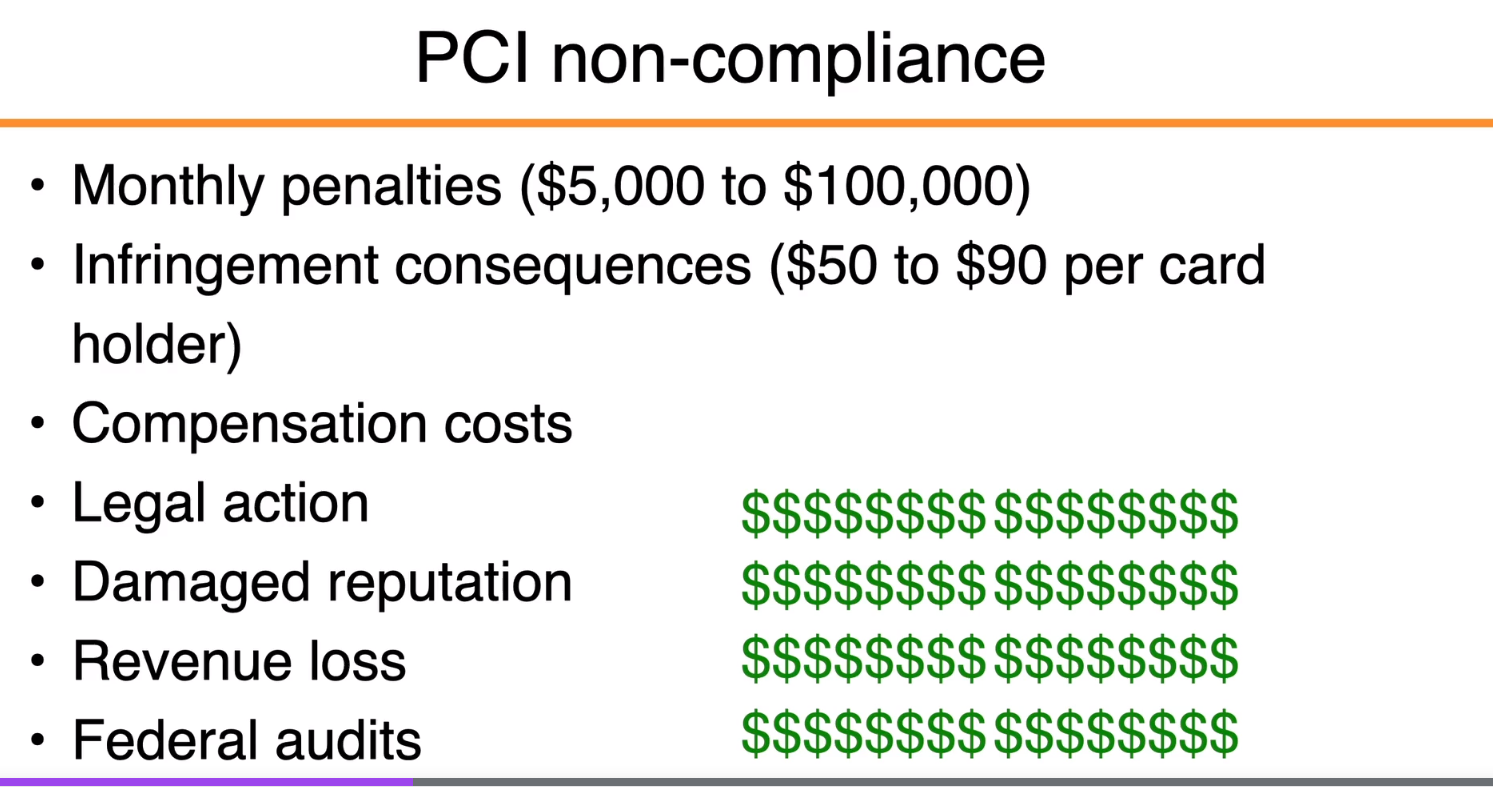
npm install react-hook-form

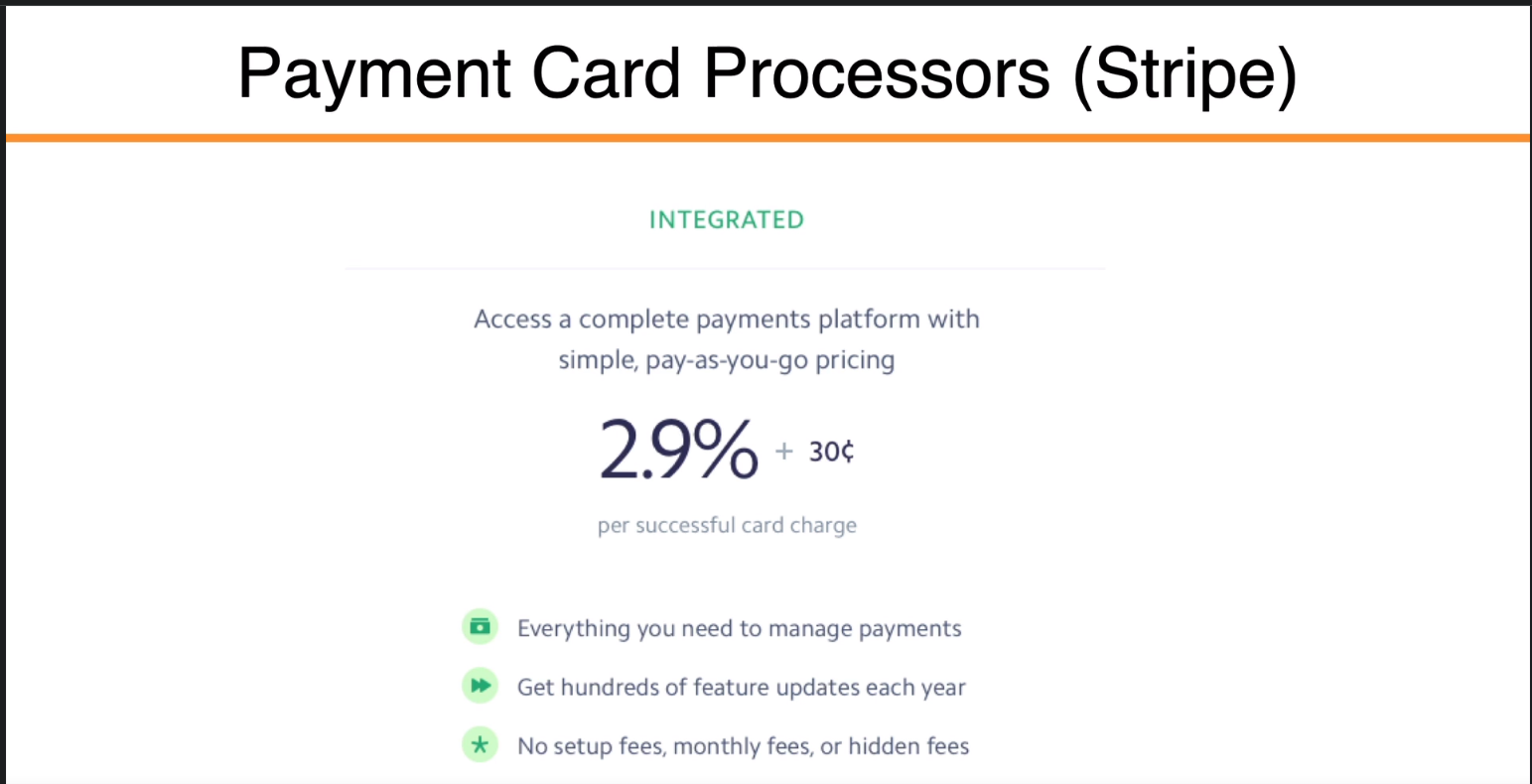
<https://react-hook-form.com/api/usecontroller>

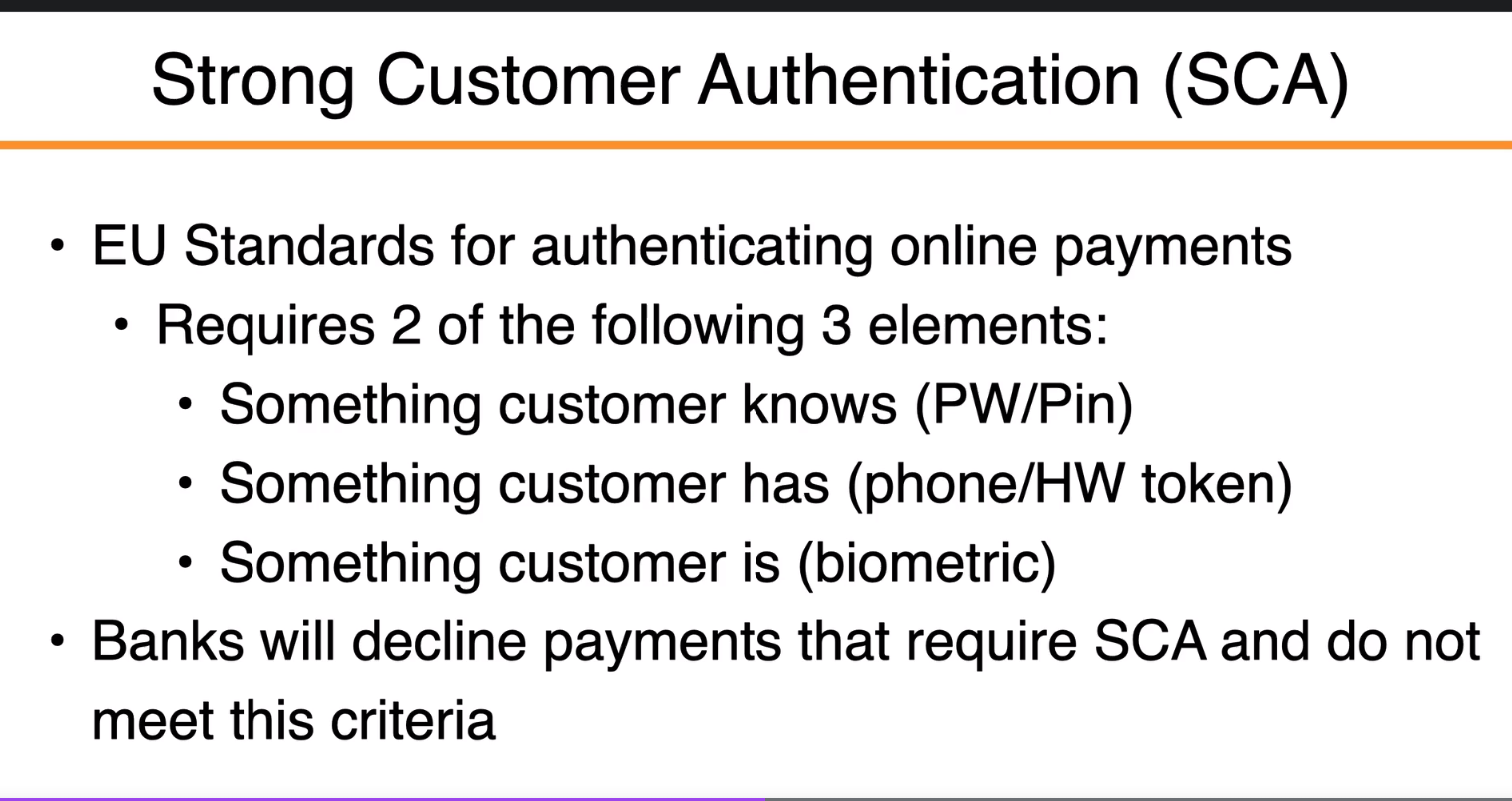


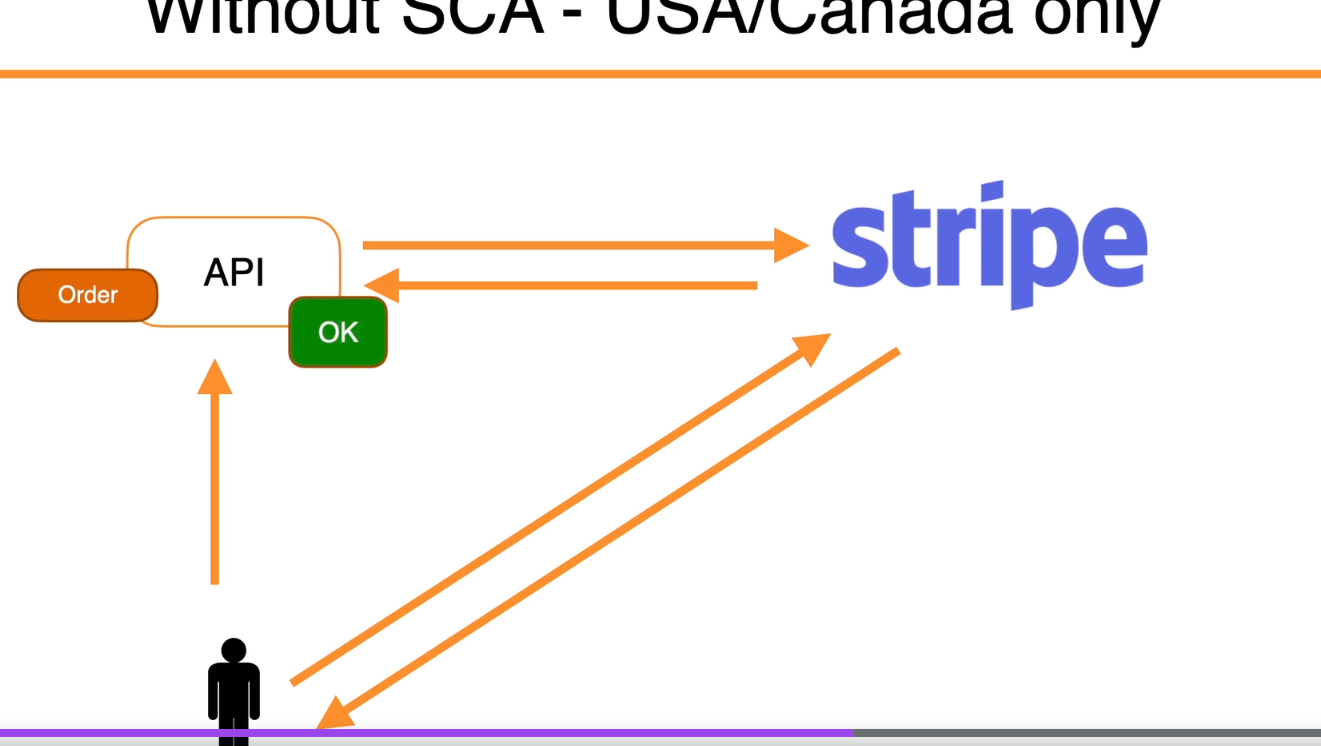


****

****

****

****

****

[**https://stripe.com/**](https://stripe.com/)

[**https://stripe.com/docs/stripe-js/react**](https://stripe.com/docs/stripe-js/react)

[**https://stripe.com/docs/testing**](https://stripe.com/docs/testing)

**npm install --save @stripe/react-stripe-js @stripe/stripe-js**

**imperative handle:**

[**https://reactjs.org/docs/hooks-reference.html#useimperativehandle**](https://reactjs.org/docs/hooks-reference.html#useimperativehandle)

**webhooks :**

**.\stripe listen -f http://localhost:5003/api/payments/webhook -e charge.succeeded**

DoTNET SECRET MANAGER:

<https://docs.microsoft.com/en-us/aspnet/core/security/app-secrets?view=aspnetcore-6.0&tabs=windows>

dotnet user-secrets init

dotnet user-secrets set "StripeSettings:PublishableKey" "pk\_test\_51HIv6eDRd0LwUKYJuG37UvfhAkLPaYAp2h2wJilEn66FfwzrXCWvojIUYeZ7gYZwVaZgLSPrOmL4YHlexcXc09ud00rhTlHFqT"

C:\Users\MosheG\AppData\Roaming\Microsoft\UserSecrets

React carousel:

<https://react-slick.neostack.com/>

npm install react-slick @types/react-slick slick-carousel

react customable environment variables:

<https://create-react-app.dev/docs/adding-custom-environment-variables/>

<https://create-react-app.dev/docs/adding-custom-environment-variables/#adding-development-environment-variables-in-env>

run in production mode:

npm run build

cloud image service:

<https://cloudinary.com/>

Assigned Cloud Name: **webtargetcloud**

React dropzone :

<https://react-dropzone.js.org/>

<https://github.com/react-dropzone/react-dropzone>

npm install --save react-dropzone

Microsoft database providers:

<https://docs.microsoft.com/en-us/ef/core/providers/?tabs=dotnet-core-cli>

production db - docker:

<https://www.docker.com/products/docker-desktop>

docker run --name dev -e POSTGRES\_USER=appuser -e POSTGRES\_PASSWORD=secret -p 5432:5432 -d postgres:latest

Heroku:

Install the Heroku cli:

<https://devcenter.heroku.com/articles/heroku-cli#install-the-heroku-cli>

heroku login

heroku git:remote -a webtarget-restore-course

Heroku buildpacks:

<https://elements.heroku.com/buildpacks>

<https://elements.heroku.com/buildpacks/jincod/dotnetcore-buildpack>

dotnet stable :

heroku buildpacks:set jincod/dotnetcore

PUSHING TO HEROKU:

git push heroku main