Creating SAFE project

- `dotnet new -i SAFE.Template`
- `dotnet new SAFE --Fulma landing --Remoting --Docker --language F#`
- `code .`, then * `build run`

Building form

- remove CSS padding for subtitle
- replace title with "SAFE Demo", subtitle with "Score my talk @...", remove `safeComponents`
- add event image (Level -> item -> Image -> img -> Src), scale it to 64x64
- remove contents of `containerBox` and `show` function
- open `Fulma.Elements.Form`, add field helper function
- add comment (Textarea) and name (Input.text)
- add submit (Button.a), make it primary color + full width
- add scores field: Level (ismobile) -> column item -> button.a -> `Icon.faIcon [ ] [ Fa.icon Fa.I.SmileO ]`
- add 2x (`Fa.fa2x` to contents), color and outlined to button
- add function `scoreIcon` - play with icons, add function `scoreColor` with `IsWhite` for no `yield`

Client side debugging

- change Model, Msg, init and update
- `Fable.Core.JsInterop`, `let onInput action = OnInput (fun e -> action !!e.target?value)`
- bind comment, name and score
- demonstrate client side debug - console, HMR, redux, react

Talking to server side

- add `Submit` to Msg, add `Loading` to Model, init, update
- bind submit button, disable all inputs when loading
- move `Score` to Shared, add Vote and VotingProtocol types
- Server: `let votes = System.Collections.ConcurrentBag<Vote>()`
- add `countVotes function`, `vote` async function with 1000 sleep
- server adapter: counter -> voting, client proxy: counter -> voting
- add `mkVote` function, `GotResults` to Msg, update - just `loading = false`
- add cmd | Submit `Cmd.ofAsync Server.api.vote (mkVote model') (Ok >> GR)...`
- add `Results` of `Result<VotingResults,exn>` to model, init
- distinguish GotResults Ok and rest in update
- `resultsBox` (empty), `formBox` and `containerBox` with pattern match
- fill out `resultsBox` -> copy from scores, but div instead of button
- add contents (small) for comments (quotes in italics)

Deploy

- build.fsx: change docker user and image name, copy image name
- Copy and adjust Deploy target (push imageFullName, add to chain)
- `build deploy` fast!
- create repository in docker hub, create web app in azure