

Упражнение: Консумиране на REST API

Messages

След като в последното занятие си създадохме Web API за съобщения, сега е време да имплементираме и Front-End частта.

Ще ни бъде предоставена обикновена HTML страница, оформена с Bootstrap. Страницата е конструирана като Front-End частта на приложението Messages. Съдържа проста форма за избор на текущото потребителско име и проста форма за изпращане на съобщения. Също така има списък с изпратените съобщения, които са всички съобщения, които в момента са в базата данни на приложението.

The screenshot shows a web interface for 'Chat-Inc.'. At the top, it says 'Welcome to Chat-Inc. A simple messenger, to do you good.' Below this is a section titled 'Choose your username to begin chatting!' which contains a text input field labeled 'Username' and a 'Choose' button. To the right of this section is a list of messages: 'Harry: Hello Mate!', 'Harry: Hello Bro!', 'Tony: Yo!', and 'Tony: What is up!'. At the bottom of the interface is a text input field for sending messages and a 'Send' button.

Не е необходимо да пишем какъвто и да е CSS. Въпреки това ще получим файл **app.js**, който трябва да допишем. Уеб API-то ни връща JSON обекти и нашата задача е да ги рендерираме на страницата с JavaScript.

Username Функционалност

След избора на потребителско име (натискане на бутона **[Choose]**) следва да се появи следният изглед

This screenshot shows the same web interface as the previous one, but after the 'Choose' button has been clicked. The 'Username' input field now contains the text 'Harry'. The 'Choose' button has been replaced by a 'Reset' button. The message list on the right remains the same: 'Harry: Hello Mate!', 'Harry: Hello Bro!', 'Tony: Yo!', and 'Tony: What is up!'. The message input field and 'Send' button at the bottom are also present.

След като кликнем върху **[Reset]**, Потребителското име трябва да бъде нулирано и да можем да изберем друго Потребителско име.

Messaging Функционалност

След натискане на бутона **[Send]** съобщението трябва да бъде изпратено до уеб API-а и то да бъде създадено в базата данни. Всички съобщения трябва да бъдат обновени (изброени отново), така че новото съобщение да може да бъде прикачено.

Micro-Validations

Нека въведем микро-валидации както следва:

- Не трябва да можем да изпращаме съобщение, без първо да сме избрали потребителско име
- Не трябва да можем да изберем празно потребителско име
- Не трябва да можем да изпращаме празно съобщение

Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма **"Обучение за ИТ кариера"** на МОН за подготовка по професия "Приложен програмист".



- Курсът е базиран на учебно съдържание и методика, предоставени от фондация **"Софтуерен университет"** и се разпространява под **свободен лиценз CC-BY-NC-SA** (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).

