Изоморфные приложения на React и Server Side Rendering

№ урока: 8 **Курс:** React Advanced

Средства обучения: Текстовый редактор или IDE, браузер, Node.js, терминал

Обзор, цель и назначение урока

В этом уроке вы познакомитесь с принципом построения изоморфных приложений с использованием React, узнаете об их основных преимуществах и научитесь создавать собственные изоморфные приложение применяя технику Server Side Rendering.

Изучив материал данного занятия, учащийся сможет:

- Знать об основных принципах и преимуществах изоморфных приложений
- Уметь создавать собственные изоморфные приложения
- Применять технику SSR (Server Side Rendering)

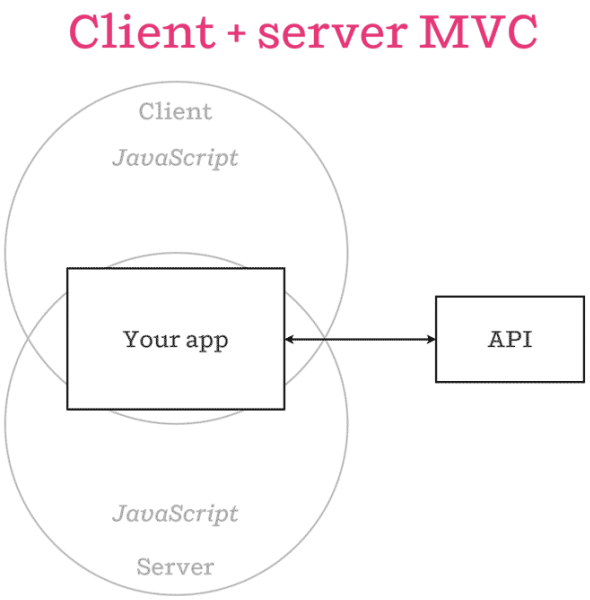
Содержание урока

1. Что такое изоморфизм?
2. Основные преимущества изоморфных приложений
3. Создание изоморфного приложения используя Server Side Rendering на Express.js

Резюме

Изоморфизм - это довольно обширное понятие, используемое во многих науках, и если вкратце, то оно означает отношение между какими-либо объектами, выражающее схожесть их структуры.

Для нас же изоморфизм означает то, что написав приложение единожды - мы можем его запускать на разных платформах - как на клиенте, так и на сервере - избегая при этом дублирования кода.



CyberBionic systematics

CyberBionic Systematics ® 2016
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

И сейчас существует довольно много холиваров, по-поводу того, как же все-таки правильно называть такие приложения – Изоморфные или Универсальные?

Но по-сути оба этих термина об одно и том же.
Суть заключается в том, что приложения на client-side требуют время для получения ответа от сервера, скачивание всех необходимых скриптов и выполнение их на клиенте (что может быть тоже длительным процессом). Соответственно за счёт такой медлительности мы получаем плохой общий UX.
Другая проблема заключается в том, что поисковые роботы пока что лишь частично умеют парсить JavaScript и парсить из него контент, поэтому помимо выше сказанного вы еще и получаете плохую индексацию вашего сайта.

Поэтому есть тенденция использовать гибридный подход: мы получаем сформированную HTML-разметку для быстрой начальной загрузки сайта и хорошей индексации роботами, а далее уже подгружаем наш JavaScript, с помощью которого «оживляем» наше приложение, делая его гибким и быстрым.

Резюмируя, вот какие выгоды мы получаем используя такой гибридный подход:

1) SEO
2) Быстрая начальная загрузка
3) Лучший общий UX
4) Прогрессивное улучшение / изящная деградация (fallbacks)

+ отдельным пунктом можно выделить 'maintainability' – за счёт того, что мы большую часть нашего кода переиспользуем, как клиентом, так и сервером.

Закрепление материала

- Назовите способ, с помощью которого можно рендерить React-компоненты на сервере?
- Каким образом осуществляется механизм progressive enhancement / graceful degradation?
- Можно ли применять Server Side Rendering на других платформах, кроме Node.JS?

Самостоятельная деятельность учащегося

Попробуйте реализовать раутинг на стороне сервера, за счёт которого сервер будет вам отдавать контент на нужной странице.

Рекомендуемые ресурсы

Изоморфный JavaScript — будущее веб-приложений
https://habrahabr.ru/post/203444/

Изоморфное Приложение с React и Redux
https://habrahabr.ru/post/264423/

Server-Side Rendering with Redux and React-Router
https://www.codementor.io/reactjs/tutorial/redux-server-rendering-react-router-universal-web-app

Server-side React Rendering: Isomorphic JavaScript with ReactJS + Node
https://reactjsnews.com/isomorphic-javascript-with-react-node

Подборка различных мануалов и примеров по SSR
https://github.com/enaqx/awesome-react#server-side-rendering

Server side rendering with React and Express (пример)
https://medium.com/front-end-hacking/server-side-rendering-with-react-and-express-382591bfc77c

CyberBionic systematics

CyberBionic Systematics ® 2016
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

t. +380 (44) 361-8937
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com