**„Fullstack React”**

**app 01) Product**

- First component example (s35), ReactDOM.render (s43), child component (s44), data driving, seeding (s46), sending props to child (s48), child receivig props (s49), rendering multiple items, map (s51), o kluczach słów kilka (s53), sortowanie (s54), funkcje do komunikacji z rodzicem (s56), binding this, constructor (s59), using state (s61), update-push (s65)

**app 02) Time logging**

STEP1?, STEP2: static version of the app, ternary operator (s94), STEP3: what should be stateful? (s98), STEP4: where pieces of state should live? (s100), hard-code initial states (s102), onChange funkcji (s109), form elements (s110), STEP6: inverse data flow (s110), server version (s122)

**app 03) Virtual DOM**

DOM root reference typy (s157), create element (s161), cond. expression (s164), bool attributes (s164), htmlFor – formy (s166), emoji JavaScript escape (s167),

**04) Advanced component config**

render (s172), props (174), state (s182), props.children (s196), React.Children, forEach (s199)

**app 05) Forms**

events and event handlers (s205), multiple fields (s216), validation (s221), **redux – do przerobienia** (od s 234),

**06) Webpack (do przerobienia)**

**07) Unit testing**

**08) Routing**

**09) Intro to Flux and Redux (do przerobienia)**

**2) „React w działaniu”**

*- Start: instalacja klienta - https://react-cn.github.io/react/downloads.html (****s17****). Przykład (****s18****).* document.getElementById (**s21**). *React.DOM (****s21****).Parametrymetod (****s22****). Class, For, Stye (****s25****). Dev Tools (****s26****). Tworzenie komponentów (bloków) (****s30****). Właściwości (dodawanie treści) (****s31****).* getDefaultProps() – domyślne właściwości (**s35**). Stan – dane na podstawie których komponent się renderuje (**s36**). getInitialState - stan(**s36, s40**). Obszar tekstowy (**s37**). \_textChange (**s40**). Dostęp do komponentu z zewnątrz – lepiej nie korzystać (**s45**). Zmiana właściwości w locie (**s47**). Metody cyklu życia – do nasłuchiwania zmian w komponencie przed i po zmianach (**s48**). Mix (domieszka) logów (**s53**). Komponent potomny (**s54**). Zapobieganie aktualizacjom (jak nic się nie zmienia) (**s56**).

- Tabelka: Metoda tablicowa Map() (**s65**). Nazwy komponentów i debugowanie (**s66**). arrayOf – typ elementów i PropTypes – weryfikacja danych komponentu (**s70**). Obsługa kliknięcia i sortowanie (**s71**). Edycja danych (**s74-s78**). Edycja tylko jednej komórki (**s79**). Wyszukiwanie (**s80-86**). Przywracanie stanu (**s87**). Pobieranie danych z pliku (**s88**).

- JSX: Początek (**s91**). Babel (**s93**). Link klienta Babel (**s93**). Białe znaki (**s100**). Komentarze (**s101**). Ucieczki XSS (**s104**). Atrybut rozszczepiania – uniknięcie ponownej definicji właśiwości (**s105**). Zwracanie kilku węzłów (**s106-s108**). Składnia (**s109**). Formularze (**s110**). <textarea> (**s111**). <select> (**s113**).

- Konfiguracja i budowanie: Foldery (**s116**). Moduły (**s120**). Instalacja pakietów (**s123**). Budowanie (**s127**). Budowanie podczas rozwijania aplikacji (**s127**). Pakiet Dev Watch (**s127**). Wdrożenie (**s128**). package.json (**s175**). Babel i scripts (**s176**). ESLint – analiza kodu (**s177-s179).** Flow – weryfikacja typów statycznych (**s180-s190**). Testowanie (**s190-s206**).

- Flux (komunikacja między komponentami): do ogarnięcia, może kiedyś.

- Przykładowa aplikacja: Struktura plików i folderów (**s132**). Import i export komponentów (s133).