**1. Originalaus PDF doko loadinimas iš nuorodos. // sukurtas loadPDFContent.mjs**

Šiuo metu loadinimas vyksta iš vienos nuorodos.

**Klausimai:**

* Ar ar gali pabūti loadinama iš skirtingų nuorodų? Nes pagal galimus (siūlomus) šablonus matau, kad viename puslapyje gali būti ir shipping label, ir packing slip. ir juos spėju gausime skirstingų sourcų?

**Tolimesni darbai. (pažymėti yes/no)**

* Papildyti loadinimą funkciją iš skirtingų nuorodų. Padaryti bent iš dviejų sourcų. loadPDFContent.mjs yes/no

**2. Loadinto dokumento apkarpymas. // sukurtas crop.mjs**

-----------------------------------------------------------------------------------------------------------------

Originalaus dokumento apkarpymas su Pdf-lib. Naudosime *mediaBox,* dokumento parametrų keitimui. <https://pdf-lib.js.org/docs/api/classes/pdfpage#setmediabox>

-----------------------------------------------------------------------------------------------------------------

**Pastaba.**

Šiuo metu iš loadinto dokumento yra paimamas tik konkretus lapas, pagal visą logiką kiek suprantu reikės paimti visus orginalaus vienos ar keleto dokumento lapus ir embedinti į naujai sukurtą pdf dokumentą.

**Klausimas:**

* Ar gali būti toks variantas, kad reikės orginaliame dokomente apkarptyti skirtingus dokumento lapus, skirtingu formatu. Pvz. turime PDF dokumentą, kuriame yra 6 lapai. Pirmame lape label turi būti 200px x 400px, antrame 150px x 150px, trečio lapo formatas kaip pirmojo, ketvirto, kaip antrojo ir t.t.?
* Ar gali būti, kad sourcinio PDF nereiks apkarpyti, o tik embedinti į naują PDF dokumentą?

**Tolimesni darbai. (pažymėti yes/no)**

* Papildyti apkarpymą iš skirtingų sourcų gautiems dokumentus. Plius galimybė apkarpyti dokumentus skirtinguose lapuose. crop.mjs yes/no

**3. Naujo dokumento sukūrimas su nustatyto dydžio/formato lapais (valdomas per parametrus).**

-----------------------------------------------------------------------------------------------------------------

Dokumento sukūrimas

const pdfDoc = await PDFDocument.create()

Page pridėjimas:

<https://pdf-lib.js.org/docs/api/classes/pdfdocument#addpage>

-----------------------------------------------------------------------------------------------------------------

**Pastaba.** Šiuo metu, kai pridedamas vienas naujas lapas viskas aišku, tačiau kaip lapų > 1 reikės papildomai pagalvoti kaip tinkamai padaryti. Teoriškai naujų lapų dokumente turės būti sukurta tiek, kiek nurodysim embadinti orginalių (apkarpytų lapų). Tai iš kiekis tokiu atveju neturėtų būti nurodomas per parametrą, o išsiskaičiuoti pagal embedinamus lapus.

3.1. Valdymas per parametrus:

3.1.1. pageSize nustatymas:

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/#const-pagesizes>

-----------------------------------------------------------------------------------------------------------------

3.1.2. Spausdimo area nustatymas (per parametrą):

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfpage#settrimbox>

-----------------------------------------------------------------------------------------------------------------

3.1.3. Atstumo tarp tarp mediaBox ir TrimBox keitimas dokumente (per parametrą):

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfpage#getbleedbox>

-----------------------------------------------------------------------------------------------------------------

3.2. Atrinkimas atskirų lapų iš originalas dokumento (ų), kurie bus keliame į naują PDF'ą. Gali reikėti imti tik atskirus lapus arba viso dokumento lapus.

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfdocument#getpages>

-----------------------------------------------------------------------------------------------------------------

**Pastaba.** Šiuo metu, kai atrenkami atskiri lapai iš source dokumetų viskas aišku, reikės išsiaiškinti kaip paiimti visus lapus ir juos įterpti į naują PDF. Prieš tai reikėtų susikaičiuoti kiek tokių sourcinių lapų buvo, tam yra kitą funcija PDF-lib, kad žinoti kiek naujų (tuščių) lapų pasiruošti, be to, šio skaičiaus gali reikėti tolimesiam embedintų dokumentų pozinionavimui.

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfdocument#getpagecount>

-----------------------------------------------------------------------------------------------------------------

3.3. Dokumentų embedinimas.

Galimi embedinimo variantai:

3.3.1. Atskirtus PDF dokumento lapus:

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfdocument#embedpdf>

-----------------------------------------------------------------------------------------------------------------

3.3.2. Atskirtus PDF lapus:

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfdocument#embedpage>

-----------------------------------------------------------------------------------------------------------------

3.3.3. Multiple PDF lapus:

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfdocument#embedpages>

-----------------------------------------------------------------------------------------------------------------

**Pastaba**. Dokumentų embedinimą PDF-lib dar reikia testuotis, nes pagal aprašymą skirtumo tarp 3.3.1 ir 3.3.2-3.3.3 nematau.

3.4. Embedinto dokumento (-ų) pozicionavimas.

-----------------------------------------------------------------------------------------------------------------

<https://pdf-lib.js.org/docs/api/classes/pdfpage#drawpage>

-----------------------------------------------------------------------------------------------------------------

**Pastaba.** Su šia vieta, taip pat matau nemažai darbo, kaip tinkamai pasiruošti embedintų dokumetų pozicionavimą. Visa tai būtų logiška vykdyti atskiroje funkcijoje, nes pagal užduotį matau, kad variantų gali būti visokių.

**Klausimas.** Ar kuriant atskirą funkciją turėtume rištis prie šablonių pozicinavimo pavyzdžių, kuriuos buvo įkėlęs Marius <https://app.asana.com/0/inbox/1200067592804698> ?

Šiuo atveju šablonus reikėtų kažkaip pavadinti, kad priklausomai nuo konkretaus šablono pasirinimo būtų vykdomas embedintų dokų pozicinionamas. Be to, manau reikia numatyti, kad būtų galimą pridėti lengvai naują šabloną. Tokiu atveju turėtume atskirą failą su daugiau nei viena funkcija.

**Tolimesni darbai. (pažymėti yes/no) //** *darbus aprašysiu kai išsigryninsime judėjimo kryptį.*

\*\*Naudingos nuorodos

<https://thecodebarbarian.com/working-with-pdfs-in-node-js.html>

<https://www.youtube.com/watch?v=Wn8d2COkr98>