

Vejledning til implementation af Publizon Reader

I. Revision

2013-04-12 – V 1.0: Oprettet.

2013-04-18 – V 1.1: Reader v1.31, nye indstillinger tilføjet.

2013-05-07 – V 1.2: Scheme-forklaringer opdateret.

2013-07-24 – V 1.3: Rettelser i scheme-forklaringer.

2013-08-26 – V 1.4: Opdatering til reader version 1.3.3

2013-11-17 – V 1.5: Opdatering til reader version 1.3.6

2014-05-28 – V 1.6: Opdatering til reader version 1.4.0, som indeholder væsentlige ændringer i måden, hvorpå indhold ombrydes.

2014-06-11 – V 1.7: Opdatering af funktionsnavne i reader version 1.4.0.

2014-06-11 – V 1.8: Opdatering af scheme-kald i reader version 1.4.0.

2014-08-20 – V 1.9: Opdatering til reader version 1.4.4.

2014-09-10 – V 1.10: Nye indstillinger tilføjet.

2014-11-14 – V 1.11: Referencer til js-filer opdateret.

2014-11-17 – V 1.12: Sætningen "Såfremt det vælges at benytte medfulgte default styling / css, SKAL id'et være "reader-container"" tilføjet i beskrivelsen til [elementId]

II. Introduktion

Publizon tilbyder en online og offline browser- og streamingbaseret Reader med hvilken slutbrugeren kan læse e-bøger. Readeren er baseret på javascript, html og css.

III. Sådan tilføjes Readeren på en side

Indsæt følgende referencer på den side, hvor readeren ønskes vist:

```
<link rel="stylesheet" type="text/css" href="https://reader.pubhub.dk/scripts/reader-1.4.4/css">
<script src="https://reader.pubhub.dk/scripts/reader-1.4.4/js" type="text/javascript"></script>
```

Aktivér Readeren med følgende kode:

```
Reader.init({
  elementId: '[elementId:string]',
  orderId: '[orderId:string]',
  sessionKeyUrl: '[sessionKeyUrl:string]',
  isbn: '[isbn:string]',
  offlineMode: [offlineMode:boolean],
  packageBasePath: '[packageBasePath:string]',
  imageBasePath: '[imageBasePath:string]',
  menuBarEnabled: [menuBarEnabled:boolean], // Optional
  navigationArrowsEnabled: [navigationArrowsEnabled:boolean], // Optional
  sideButtonWidth: '[sideButtonWidth:string]', // Optional
  horizontalMargin: '[horizontalMargin:string]', // Optional
  topMargin: '[topMargin:string]', // Optional
  bottomMargin: '[bottomMargin:string]', // Optional
  pageSpacing: [pageSpacing:integer], // Optional
  progressBarBottom: [progressBarBottom:boolean], // Optional
  progressTextBottom: [progressTextBottom:boolean], // Optional
  flickEnabled: [flickEnabled:boolean], // Optional
  slideEnabled: [slideEnabled:boolean], // Optional
  swipeResistanceEnabled: [swipeResistanceEnabled:boolean], // Optional
```

Kommentar [t1]: Tilføjet i 1.10, erstatter verticalMargin.

Kommentar [t2]: Tilføjet i 1.10

```

    flickMaxDuration: [flickMaxDuration:integer], // Optional
    flickMinDistance: [flickMinDistance:integer], // Optional
    mobileSettings: {
        // Any of the previous settings to be overridden in mobile mode
        // E.g.
        // horizontalMargin: '[horizontalMargin:string]',
        // sideButtonWidth: '[sideButtonWidth:string]',
    }
};

```

Kommentar [t3]: Tilføjet i 1.10, erstatter alle "mobile"-indstillinger.

Hvor:

[elementId] er id'et på det html-element på siden (eksempelvis en <div>), hvor readeren skal vises/indsættes. Såfremt det vælges at benytte medfulgte default styling / css, SKAL id'et være "reader-container".

[orderId] er lånets ordrenummer (feltet <retailerordernumber>, som returneres af CreateLoan og GetLibraryUserOrderList på <https://libraryservices.pubhub.dk/>)

[sessionKeyUrl] er en url til et script på klientens server som skal returnere en sessionsnøgle, som fås ved at kalde Publizons streaming service med en licensnøgle (se nedenfor). Det er vigtigt, at dette script på denne url, tjekker om brugeren/låneren er logget ind. sessionKeyUrl'en må således ikke returnere en sessionsnøgle medmindre brugeren er logget ind på klientens website.

[isbn] (valgfri, standardværdi: null) er bogens ISBN-nummer, der anvendes til at hente sampleudgaven. Hvis denne værdi er sat, skiftes readeren til sample-mode, hvilket ændrer API-kaldene til at kalde sample-API'et, og fjerner bogmærke-, note- og huskefunktionerne.

[offlineMode] (valgfri, standardværdi: false) bestemmer om readeren skal køre offline-mode, hvilket skifter API-kaldene til at anvende reader://-schemet, der er beskrevet nærmere i afsnit V. *Offline mode*.

[packageBasePath] (valgfri, standardværdi: "") er adressen der indsættes før alle filnavne i offline-mode – dvs. stien til bogens udpakkede filer i offline-mode.

[imageBasePath] (valgfri, standardværdi: './images/<versionsnummer>') er stien til readerens ikoner.

[menuBarEnabled] (valgfri, standardværdi: true) fjerner menubaren hvis false.

[navigationArrowsEnabled] (valgfri, standardværdi: true) fjerner navigationspilene hvis false.

[sideButtonWidth] (valgfri, standardværdi: 100, mobiltilstand: '15%') bestemmer bredden i procent eller pixels på områderne i hver side, der skifter side hvis de trykkes på. Værdien kan sættes til mere end [horizontalMargin] hvis området skal gå ind over selve bogen. Sæt til 0 for at fjerne områderne.

[horizontalMargin] (valgfri, standardværdi: 100, mobiltilstand: 4) bestemmer sidemargen i procent eller pixels i desktop-/tablettilstand.

[topMargin] (valgfri, standardværdi: 0) bestemmer topmargen i procent eller pixels.

[bottomMargin] (valgfri, standardværdi: 0) bestemmer bundmargen i procent eller pixels.

[pageSpacing] (valgfri, standardværdi: 40) bestemmer afstanden mellem sider i pixels.

[progressBarBottom] (valgfri, standardværdi: false) flytter progress baren ned i bunden hvis true.

[progressTextBottom] (valgfri, standardværdi: false) flytter progress teksten ned i bunden hvis true. Der ændres ikke på sidens størrelse, så [bottomMargin] bør sættes til 20 eller mere hvis progress baren flyttes ned i bunden.

[flickEnabled] (valgfri, standardværdi: true) giver mulighed for at skifte side ved at lave en hurtig flick-bevægelse hvis true.

[slideEnabled] (valgfri, standardværdi: false) giver mulighed for at skifte side ved at slide den til siden hvis true.

[swipeResistanceEnabled] (valgfri, standardværdi: false) giver modstand, i form af at slide-hastigheden nedsættes ved swipe hvis true.

Kommentar [t4]: Tilføjet i 1.10

[flickMaxDuration] (valgfri, standardværdi: 200) ændrer den maksimale varighed af flick-bevægelsen i millisekunder før der skiftes side.

[flickMinDistance] (valgfri, standardværdi: 10) bestemmer den minimale flick-bevægelses distance i pixels før der skiftes side.

[mobileSettings] (valgfri) kan bruges til at overskrive hvilke som helst indstillinger i mobiltilstand. Oplagte muligheder er her topMargin, bottomMargin, horizontalMargin og sideButtonWidth, der kan defineres her for at bruge andre værdier i mobiltilstand.

OBS: Alle indstillinger vil overskrive eventuelle mobil-specifikke standardindstillinger. Hvis der ønskes at der skal bruges forskellige værdier i hhv. tablet/desktop-tilstand og mobiltilstand, skal indstillingerne også defineres i [mobileSettings] med de mobil-specifikke værdier.

Kommentar [t5]: Tilføjet i 1.10.

OBS: [sessionKeyUrl] og [isbn] benyttes kun i online-mode (ikke offline-mode).

IV. Sådan hentes en sessionsnøgle (kun online-mode)

En sessionsnøgle hentes via følgende url:

`https://streaming.pubhub.dk/publicstreaming/v1/{retailerOrderNumber}/session/{clientKey}/{libraryId}/{libraryKey}/`

Denne url returnerer følgende:

```
<Session
xmlns="http://streaming.pubhub.dk"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"><ContentOrderId>ac964509-479c-48ee-9988-ae902de2fcd6</ContentOrderId><Expires>2013-01-25T17:50:07.436125+01:00</Expires><SessionId>d5b9ff94-6611-4657-a17a-a730cd3ed4bd</SessionId></Session>
```

...som server-side scriptet blot skal returnerer tilbage i sin helhed.

Her følger et simpelt eksempel (skrevet i C#) på et server-side script, som kalder (og returnerer) en sessionsnøgle:

```
if (user.IsLoggedIn) {
    var client = new WebClient();
    var str =
        client.DownloadString("https://streaming.pubhub.dk/publicstreaming/v1/.../");
    context.Response.ContentType = "text/plain";
    context.Response.Write(str);
}
```

Dette bruger readerens javascript-kode så som autorisation. Readeren kalder kun denne url, hvis den ikke har en sessionsnøgle, eller hvis den er udløbet.

V. Offline mode

Online og offline mode understøttes i ét og samme javascript.

Som default er readeren i online mode. Hvis der er behov for at starte i offline mode, skal `offlineMode`-indstillingen i Readers constructor sættes til `true`.

Kald `Reader.Communication.enableOfflineMode()` hvis der skal skiftes til offline mode efter initialisering. Kald `Reader.Communication.enableOnlineMode()` hvis der skal skiftes til online mode efter initialisering.

I offline mode sendes alle kald i scheme format til app'en i stedet for web API'et, og der forventes svar tilbage for hvert kald i JSON-format. For at sende svardata tilbage skal

`Reader.Communication.receiveData(url, jsonData)` kaldes, hvor `url` er scheme-adressen, der bruges til at identificere hvor data kommer fra, og `jsonData` er en string med svardata. De forskellige scheme adresser og forventede returdata er beskrevet nedenfor.

Schema:	<code>reader://[orderId]/wordcount/</code> <code>[orderId] {string}</code> : Ordrens ID.
Beskrivelse:	Returnerer en liste med det totale antal af elementer der kommer før hvert kapitel og hele bogens totale antal elementer. Alle disse værdier kan hentes direkte fra index-filen.
Forventet svar:	<pre>{ "ChapterSizes": [{ "Key": KapitelIndex, "Value": StartOrdId }, { "Key": KapitelIndex, "Value": StartOrdId }, ...], "TotalWordCount": TotalAntalOrd }</pre> <i>KapitelIndex {int}</i> : Værdien af <code><entity></code> 's <code>index</code> -attribut i index-filen. <i>StartOrdId {int}</i> : Værdien af <code><entity></code> 's <code>firstwordindex</code> -attribut i index-filen. <i>TotalAntalOrd {int}</i> : Indholdet af <code><wordcount></code> -elementet i index-filen.
Eksempel på svar:	<pre>{ "ChapterSizes": [{ "Key": 1, "Value": 0 }, { "Key": 2, "Value": 1 }, { "Key": 3, "Value": 20 }], "TotalWordCount": 138159 }</pre>

Schema:	<code>reader://[orderId]/[chapter/]</code> <code>[orderId] {string}</code> : Ordrens ID. <code>[chapter] {int}</code> : Kapitlets ID.
Beskrivelse:	Returnerer indholdet af et helt kapitel samt kapitlets ID og hele bogens totale antal elementer. Kapitlets ID, filnavn og det totale antal elementer kan læses fra index-filen, og kapitlets indhold hentes fra filen, der refereres til.
Forventet svar:	<pre>{ "Index": KapitelIndex, "Source": KapitelKildekode, "TotalIndexCount": TotalAntalKapitler }</pre> <i>KapitelIndex {int}</i> : Værdien af <code><entity></code> 's <code>index</code> -attribut i index-filen. <i>KapitelKildekode {string}</i> : Al indholdet af (X)HTML-filen som <code><entity></code> 's værdi refererer til. <i>TotalAntalKapitler {int}</i> : Antallet af <code><entity></code> -noder i index-filen.
Eksempel på svar:	<pre>{ "Index": 5, "Source": "<html ...", "TotalIndexCount": 11 }</pre>

Schema:	<code>reader://[orderId]/indexes/</code> <code>[orderId] {string}</code> : Ordrens ID.
Beskrivelse:	Returnerer en liste af alle bogens kapitler.
Forventet svar:	[

	<pre>{ "Filename": FilNavn, "Index": KapitelIndex, "Title": KapitelTitel }, { "Filename": Filnavn, "Index": KapitelIndex, "Title": KapitelTitel }, ...]</pre> <p>FilNavn <i>{string}</i>: <entity>'s værdi. KapitelIndex <i>{int}</i>: Værdien af <entity>'s index-attribut i index-filen. KapitelTitel <i>{string}</i>: Værdien af <entity>'s title-attribut i index-filen.</p>
Eksempel på svar:	[{"Index":1, "Title":"Forside"}, {"Index":2, "Title":"Indholdsfortegnelse"}]

Scheme:	<pre>reader://[orderId]/state/[chapter]/[element]/[node]/[character]/</pre> <p><i>[orderId]</i> <i>{string}</i>: Ordrens ID. <i>[chapter]</i> <i>{int}</i>: Kapitlets ID. <i>[element]</i> <i>{int}</i>: ID på element på den nuværende side. <i>[node]</i> <i>{int}</i>: Indeks på elementets tekstnode. <i>[character]</i> <i>{int}</i>: Indeks på karakteren i tektnoden.</p>
Beskrivelse:	Gemmer brugerens nuværende læseposition. Når brugeren næste gang åbner bogen sendes han/hun til denne gemte læseposition.
Forventet svar:	<pre>{"Success": Success}</pre> <p><i>Success</i> <i>{bool}</i>: true hvis læsepositionen blev gemt, false hvis den ikke kunne gemmes.</p>
Eksempel på svar:	{"Success": true}

Scheme:	<pre>reader://[orderId]/bookmark/[chapter]/[element]/[node]/[character]?message=[message]</pre> <p><i>[orderId]</i> <i>{string}</i>: Ordrens ID. <i>[chapter]</i> <i>{int}</i>: Kapitlets ID. <i>[element]</i> <i>{int}</i>: ID på element på den nuværende side. <i>[node]</i> <i>{int}</i>: Indeks på elementets tekstnode. <i>[character]</i> <i>{int}</i>: Indeks på karakteren i tektnoden. <i>[message]</i> <i>{string}</i>: En lille del af teksten på siden hvor bogmærket blev sat.</p>
Beskrivelse:	Gemmer et bogmærke på brugerens nuværende læseposition. En markeringsidentifikation returneres, der bruges til at identificere bogmærket hvis det senere skal slettes.
Forventet svar:	<pre>{"MarkingId": MarkeringId}</pre> <p><i>MarkingId</i> <i>{string}</i>: ID på bogmærket der blev gemt.</p>
Eksempel på svar:	{"MarkingId": "10a77382-22cf-461c-9f70-8c4ff92fe8fa"}

Scheme:	<pre>reader://[orderId]/note/[chapter]/[element]/[node]/[character]/[message]/</pre> <p><i>[orderId]</i> <i>{string}</i>: Ordrens ID. <i>[chapter]</i> <i>{int}</i>: Kapitlets ID. <i>[startElement]</i> <i>{int}</i>: ID på første element på siden. <i>[endElement]</i> <i>{int}</i>: ID på sidste element på siden. <i>[message]</i> <i>{string}</i>: Brugerens indtastede note.</p>
Beskrivelse:	Gemmer en note på brugerens nuværende læseposition. En markeringsidentifikation returneres, der bruges til at identificere noten hvis den senere skal ændres eller slettes.
Forventet svar:	{"MarkingId": MarkeringID}

	<i>MarkeringId</i> {string}: ID på noten der blev gemt.
Eksempel på svar:	<code>{"MarkingId": "e8f9068d-e330-4065-b39a-e0f49fc8c21b" }</code>

Scheme:	reader://[orderId]/marking/update/[markingId]/[message]/ <i>[orderId]</i> {string}: Ordrens ID. <i>[markingId]</i> {string}: Notens ID. <i>[message]</i> {string}: Brugerens indtastede note.
Beskrivelse:	Ændrer noten med den givne markeringsidentifikation.
Forventet svar:	<code>{"Success": Success}</code> <i>Success</i> {bool}: true hvis noten blev opdateret, false hvis den ikke kunne opdateres.
Eksempel på svar:	<code>{"Success": true}</code>

Scheme:	reader://[orderId]/marking/delete/[markingId]/ <i>[orderId]</i> {string}: Ordrens ID. <i>[markingId]</i> {string}: Notens ID.
Beskrivelse:	Sletter noten eller bogmærket med den givne markeringsidentifikation.
Forventet svar:	<code>{"Success": Success}</code> <i>Success</i> {bool}: true hvis noten/bogmærket blev slettet, false hvis den ikke kunne slettes.
Eksempel på svar:	<code>{"Success": true}</code>

Scheme:	reader://[orderId]/marking/ <i>[orderId]</i> {string}: Ordrens ID.
Beskrivelse:	Returnerer en liste med alle bogmærker og noter samt en eventuel gemt læseposition.
Forventet svar:	[{ "ContentIndex": KapitelIndex, "Definition": MarkeringsType, "MarkingId": MarkeringsId, "Message": Meddelelse, "ElementIndex": ElementIndex, "NodeIndex": NodeIndex, "StartCharIndex": CharacterIndex }, { "ContentIndex": KapitelIndex, "Definition": MarkeringsType, "MarkingId": MarkeringsId, "Message": Meddelelse, "ElementIndex": ElementIndex, "NodeIndex": NodeIndex, "StartCharIndex": CharacterIndex }, ...] <i>KapitelIndex</i> {int}: Kapitlets ID. <i>MarkeringsType</i> {string}: "Bookmark", "Note" eller "ReadingStateMark". <i>MarkeringsId</i> {string}: Markeringsidentifikationen <i>Meddelelse</i> {string}: Notens indhold hvis det er en note, tom streng hvis det er et bogmærke. <i>ElementIndex</i> {int}: Index på elementet i det givne kapitel hvor bogmærket/noten er gemt. <i>NodeIndex</i> {int}: Indeks på elementets tekstnode. <i>CharacterIndex</i> {int}: Indeks på karakteren i tekstnoden.
Eksempel på svar:	[{ "ContentIndex": 8, "Definition": "Bookmark", "MarkingId": "a7ddb11a-e554-4137-a6fa-218f79888d63", "Message": "", "ElementIndex": 5,

	<pre>"NodeIndex":0,"StartCharIndex":2}, {"ContentIndex":80,"Definition":"Note","MarkingId":"76fdb6f-7bb4-470d-9731-2ecd06048027","Message":"Vigtigt!", "ElementIndex":54, "NodeIndex":2,"StartCharIndex":44}, {"ContentIndex":20,"Definition":"ReadingStateMark","MarkingId":"76fdb6f-7bb4-470d-9731-2ecd06048027","Message":"","ElementIndex":83, "NodeIndex":0,"StartCharIndex":505}, ...]</pre>
--	---

VI. Synkronisering af data

Ved skift fra offline til online er det app'ens ansvar at sende nye/opdaterede/slettede bogmærker, noter og "huske-mærke" til API'et, så de gemmes på serveren. Se dokumentet Publizon Offline Reader App_X.Y.docx for mere info.