PDF Highlighter

# Overview

The PDF Highlighter component can be used in Appian applications, where a user wants to highlight a particular data in a PDF document under a particular label.This component not only allows the user to select the text, the end user can also select a particular area in the PDF and mark it under a particular label.

The highlights created by the user will be listed in the sidebar with details.The sidebar also gives the option to delete the highlights.The end user can move around the highlights in the PDF document by selecting the highlight in the sidebar.The sidebar label is customizable by passing it as a input to the component and you can able the hide the sidebar from the end user.

The labels of the highlights will be displayed over the highlight when the user hover over any highlight.This component can also be integrated with the Appian interface, to move around the highlights in a PDF document.

If you don’t want to limit the end user to use a certain number of topics, you can also allow the end user to enter their own custom label for the highlights.You can also customize the color of the highlights which is used in the component.

This component also comes with ‘read only’ mode. In read only mode the end user is only able to view the annotation passed as input and they won’t be able to create or delete a highlight.

This component also has the ability to display the extracted document details from Abbyy flexi capture application and Zuva as text highlight in the PDF document.

# Features

* Highlight areas or text in a PDF document.
* Display extracted document details from Abbyy flexi capture and Zuva as text highlights.

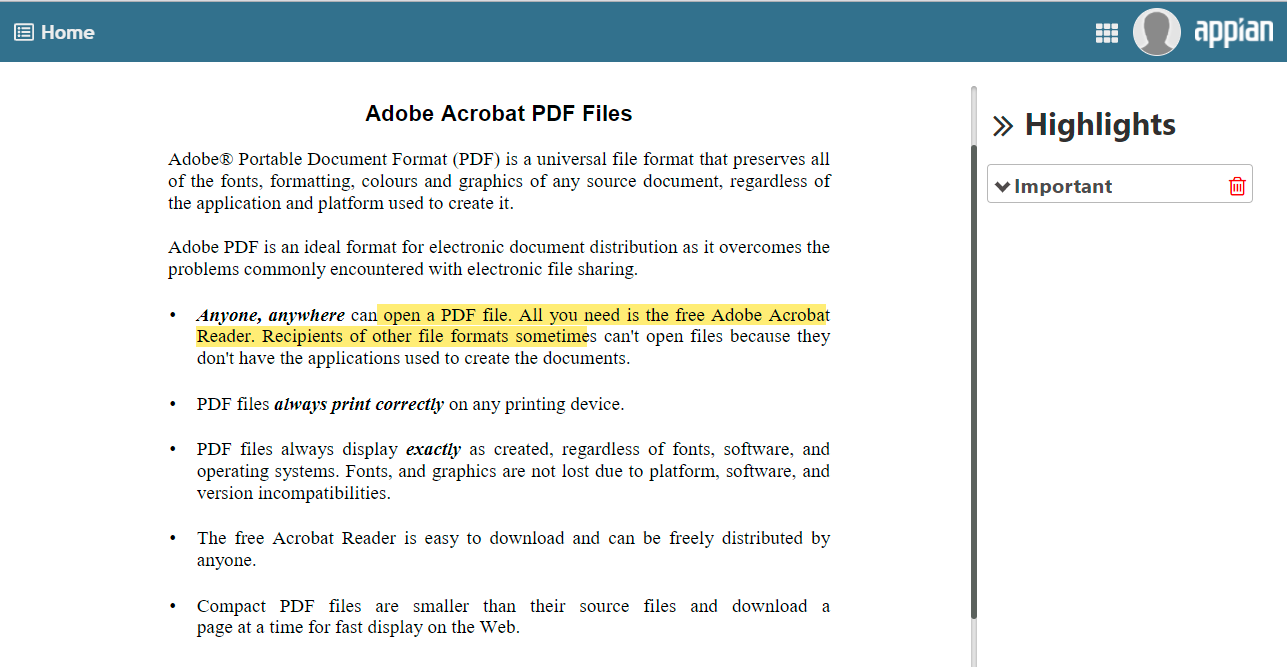
# pdfHighlighterField()

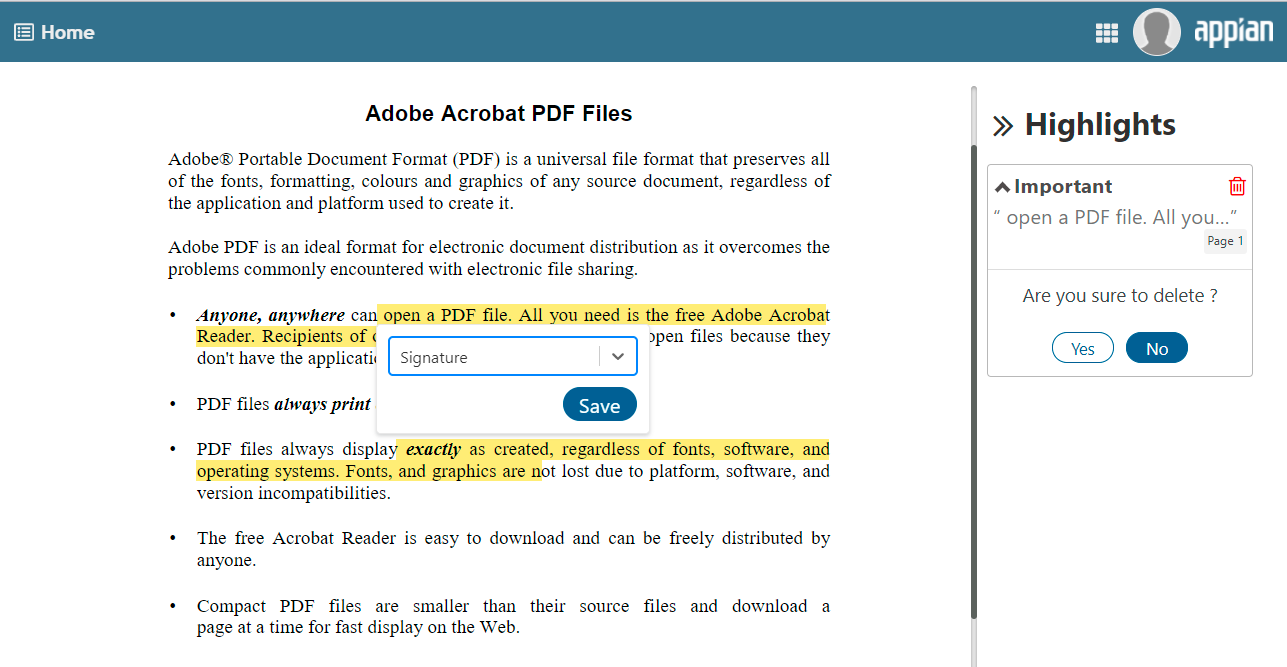
The pdfHighlighterField() allows the end user to view and create highlights in a PDF document and save it to Appian.

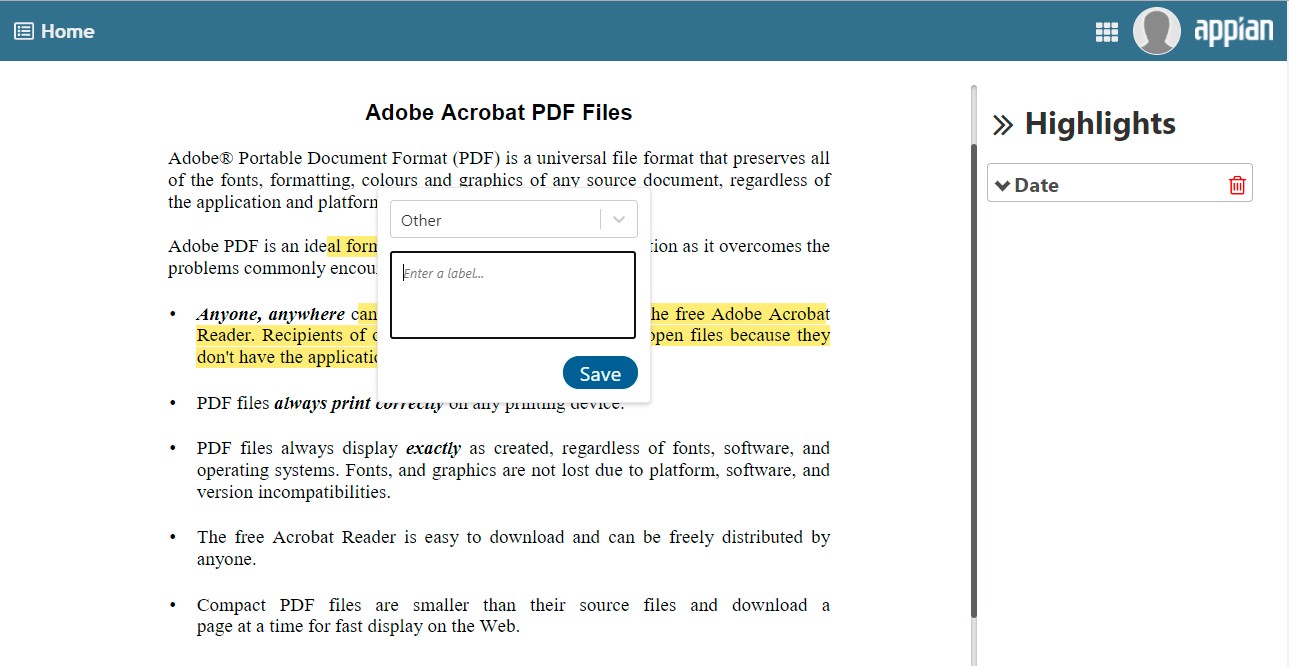
## Note

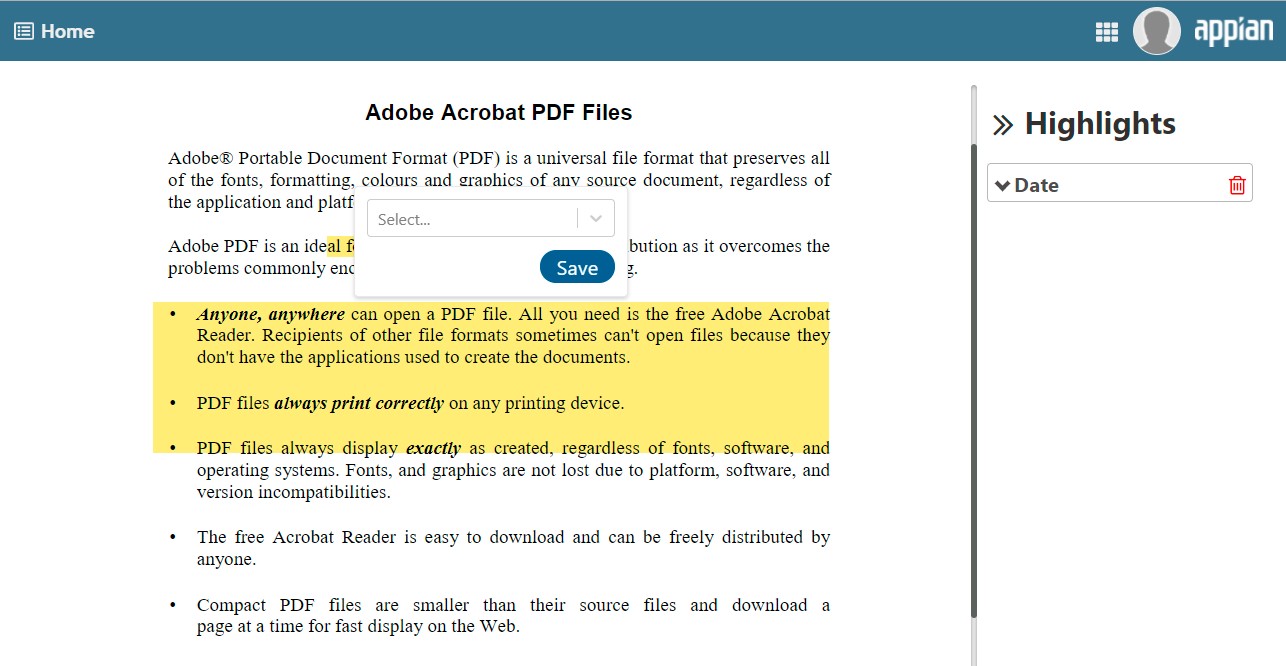
1. This component will work well for a fully digitized PDF document.
2. The user will still be able to create highlights while the sidebar is hidden, but they won't be able to delete.
3. To select a particular area in the document, press “Alt” on the keyboard and use the mouse to drag over the area you want to highlight.
4. This component needs connected system to access the PDF documents, you can get the connected system from this [link](https://community.appian.com/b/appmarket/posts/pdf-annotation-document-access-connected-system)
5. Supported Browsers - Google Chrome, Mozilla Firefox, Edge, Safari.
6. Supported Language - English (United States).
7. Best viewed in Medium,Tall,Auto (Heights)
8. Not a good fit for mobile devices

## Screenshots

****

****

****

****

## Parameters

| **Name** | **Keyword** | **Type** | **Description** |
| --- | --- | --- | --- |
| Label | label | Text | Text to display as the field label. |
| Label Position | labelPosition | Text | Determines where the label appears. Valid values: “ABOVE” (default), “ADJACENT”, “JUSTIFIED”, “COLLAPSED”. |
| Instructions | instructions | Text | Text to display as the field instruction. |
| Document ID | documentId | Number | Appian document ID of the PDF file, which will be displayed by the component |
| Connected System | connectedSystem | Connected System | Connected system to access the file from the Appian. |
| Highlights Value | highlightsValue | Array of Dictionary | Highlight’s data of the PDF document, which will appear in the component. The highlight data should be in the following format.  {  Id: “123”,  name: “Label the user selected”,  value: “Text the user selected”,  blockref:bounding reacts,  contentType: “text or image”  }  To pass the Abbyy or Zuva extraction details, please use the below structure.  Example:  {  documentExtractionData: {  {  "id": "9903",  "label": "Highlight Label",  "text": "Highlighted text",  "blockRef": {  "boundingRect": {  "left": 302,  "right": 801,  "top": 673,  "bottom": 710  },  "rects": {  {  "left": 302,  "right": 801,  "top": 673,  "bottom": 710  }  },  "pageNumber": 5  }  }  }  } |
| Enable Custom Label | enableCustomLabel | Boolean | Determines whether the user is allowed to enter their own labels for the highlights.  Values: True or False  Default : True |
| Highlight Labels | highlightLabels | Array of Text | List of labels, you want to list for the user to select for the highlights.  Example  {“Important”, “End Date”, “Name”} |
| Enable Area Selection | enableAreaSelection | Boolean | Determines whether the user is allowed to make area selection in the PDF.  Values: True or False  Default : false |
| Highlight Color | highlightColor | Text | Determines the color of the highlights created by the end user. Default YELLOW. Provide a valid 3 or 6 digit hex value to change the highlight color or choose from the predefined set YELLOW, BLUE, PINK, GREEN. |
| Highlight to Focus | highlightToFocus | Text | Specify the highlight ID, which will be focussed in the PDF document.  Example: “123” |
| Focused Highlight Color | focusedHighlightColor | Text | Determines the focus color of the current highlight in focus. Default: #49e65e. Provide a valid 3 or 6 digit hex value to change the focus color. |
| Highlights Save Into | highlightsSaveInto | Array of Dictionary | Highlight data returned from the component in th following format  Example:  [  {  Id:”12312”  Name: “Date”  Value: “19-10-2021”  blockRef:{contains the positions data}  contentType: “text”  }  ] |
| Highlight Event | highlightEvent | Dictionary | Return the last updated highlight with the updated type, in a dictionary format in the below mentioned format  {  Id: “123”  Name: “Date”  Value: “12-20-02”  blockRef:{contains the positions data}  contentType: “text” or “image” based on  highlight type  upDateType: “created” or “deleted” based on  the action performed  } |
| Read Only | isReadOnly | Boolean | Determines whether the end user is permitted to create and delete highlights in the PDF document  Values: True or False  Default : false |
| Sidebar Label | sideBarLabel | Text | Text to display as the sidebar label. |
| Hide SideBar | hideSideBar | Boolean | Determines whether the sidebar is visible to the end user.  Values: True or False  Default : false  NOTE: The user will still be able to create highlights while the sidebar is hidden, but they won't be able to delete. |