

<https://nodejs.org/en/>

```
npm outdated -g  
npm install @microsoft/generator-sharepoint@latest -g
```

```
npm install -g yo gulp  
npm install -g @microsoft/generator-sharepoint
```

```
md hellowebpart  
cd hellowebpart  
yo @microsoft/sharepoint
```

```
gulp serve
```

sprememba teksta

sprememba property pane
<http://www.sharepointnutsandbolts.com/2016/09/sharepoint-framework-web-part-properties-pages-groups-controls.html>

```
PropertyPaneCheckbox,  
PropertyPaneDropdown
```

```
prikazi: boolean;  
barva: string;
```

```
PropertyPaneCheckbox('prikazi', {  
  text: strings.PrikaziFieldLabel,  
  disabled: false,  
  checked: true  
}),  
PropertyPaneDropdown('barva', {  
  label: "Barva ozadja",  
  options: [  
    {key: 'Red', text: 'Rdeča'},  
    {key: 'Green', text: 'Zelena'},  
    {key: 'Blue', text: 'Modra'}  
  ],  
  selectedKey: 'Blue'  
})
```

```
gulp serve --locale=sl-si  
this.context.pageContext.cultureInfo.currentUICultureName
```

```
<p style={ !this.props.prikazi ? { display: 'none' } : { } } className={ styles.description  
>{escape(this.props.description)}</p>
```

```
<div style={ { backgroundColor: this.props.barva } } className={ styles.row }>
```

sprememba ikone

- fabric UI <https://developer.microsoft.com/en-us/fabric#/styles/icons>

Bus

- custom ikona

```
"iconImageUrl": "http://icons.iconarchive.com/icons/custom-icon-design/flatastic-1/512/application-icon.png",
```

Validacija

```
private descriptionValidator(value: string): Promise<string> {  
  console.log(value);  
  if (value.length > 10) {  
    return Promise.resolve("");  
  }  
  else {  
    return Promise.resolve("Description mora biti dolg vsaj 10 znakov.");  
  }  
}
```

onGetErrorMessage: this.descriptionValidator

props vs state

Can get initial value from parent Component?	Yes	Yes
Can be changed by parent Component?	Yes	No
Can set default values inside Component?*	Yes	Yes
Can change inside Component?	No	Yes
Can set initial value for child Components?	Yes	Yes
Can change in child Components?	Yes	No

```
constructor()  
componentWillMount()  
render()  
componentDidMount()
```

Header & footer app customizer

```
?loadSPFX=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&customActions={"f0362066-9a5e-4a24-8bd2-166b4875a575":{"location":"ClientSideExtension.ApplicationCustomizer","properties":{"Top":"Top area of the page","Bottom":"Bottom area in the page"}}
```

Task - percentage

```
?loadSPFX=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&fieldCustomizers={"Completed":{"id":"eb869e04-cdb4-4057-9f54-f99131022ca5","properties":{"midStart":"30","midEnd":"70"}}
```

Description field customizer

```
npm install --save jquery@1.12.4
npm install --save jqueryui@1.11.1
npm install --save sp-pnp-js
```

```
import * as jQuery from 'jquery';
import 'jqueryui';
```

```
import pnp, { List, ItemUpdateResult, Item } from 'sp-pnp-js';
import { SPComponentLoader } from '@microsoft/sp-loader';
```

```
SPComponentLoader.loadCss('//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css');
```

```
pnp.sp.web.lists.getByTitle(this.context.pageContext.list.title).items.getById(parseInt(event.row.getValueByName('ID').toString()))
    .get()
    .then(r => {
        event.cellDiv.innerHTML = '<div class="spfx-tooltip"><a href="#" title="' + r.AdditionalDescription +
">" + CellFormatter.renderAsText(this.context.column, event.cellValue) + '</a></div>';
```

```
    jQuery(function() {
        jQuery('.spfx-tooltip').tooltip();
    });
});
```

gulp serve --nobrowser

```
?loadSPFX=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&fieldCustomizers={"Description":{"id":"64256241-f4fb-4ff7-a7fe-a7a8445cf07a"}}
```

Breadcrumb

```
?loadSPFX=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&customActions={"071c  
baef-2d2a-4f4a-bd74-cebfc9de7a8a":{"location":"ClientSideExtension.ApplicationCustomizer"}}
```

Term store navigation

```
?loadSPFX=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&customActions={"df8d  
44ba-6145-467b-b411-  
7c6d3f831d30":{"location":"ClientSideExtension.ApplicationCustomizer","properties":{"NavTermSet":"T  
opNavigation","FooterTermSet":"BottomNavigation"}}
```

Planner field customizer

```
npm install --save @microsoft/sp-dialog
```

Ustvari PlannerDialog.tsx fajl

```
?loadSpfx=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&customActions={"5a16  
58e8-8ab1-4906-93b5-  
f9dbd97dd3fc":{"location":"ClientSideExtension.ListViewCommandSet.CommandBar","properties":{"pla  
nId":"AwbQG3sX9Ea-AOQHvHsqFZcADUbE"}}
```

Workbench URL: [_layouts/15/workbench.aspx](#)

Application customizer

PlaceholderContent,
PlaceholderName

AppCustomizer.module.scss

```
.app {  
  .top {  
    height:60px;  
    text-align:center;  
    line-height:2.5;  
    font-weight:bold;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
  }  
}
```

```
.bottom {  
  height:40px;  
  text-align:center;  
  line-height:2.5;  
  font-weight:bold;  
  display: flex;  
  align-items: center;  
  justify-content: center;  
}  
}
```

```
import styles from './AppCustomizer.module.scss';  
import { escape } from '@microsoft/sp-lodash-subset';
```

```
Top: string;  
Bottom: string;
```

```
private _topPlaceholder: PlaceholderContent | undefined;  
private _bottomPlaceholder: PlaceholderContent | undefined;
```

```
// Added to handle possible changes on the existence of placeholders.  
this.context.placeholderProvider.changedEvent.add(this, this._renderPlaceHolders);
```

```
// Call render method for generating the HTML elements.  
this._renderPlaceHolders();  
return Promise.resolve<void>();
```

```

private _renderPlaceHolders(): void {

    console.log('HelloWorldApplicationCustomizer._renderPlaceHolders()');
    console.log('Available placeholders: ',
this.context.placeholderProvider.placeholderNames.map(name => PlaceholderName[name]).join(', '));

    // Handling the top placeholder
    if (!this._topPlaceholder) {
this._topPlaceholder =
this.context.placeholderProvider.tryCreateContent(
    PlaceholderName.Top,
    { onDispose: this._onDispose });

    // The extension should not assume that the expected placeholder is available.
    if (!this._topPlaceholder) {
        console.error('The expected placeholder (Top) was not found. ');
        return;
    }

    if (this.properties) {
        let topString: string = this.properties.Top;
        if (!topString) {
            topString = '(Top property was not defined.)';
        }

        if (this._topPlaceholder.domElement) {
            this._topPlaceholder.domElement.innerHTML = `
                <div class="${styles.app}">
                    <div class="ms-bgColor-themeDark ms-fontColor-white ${styles.top}">
                        <i class="ms-Icon ms-Icon--Info" aria-hidden="true"></i> ${escape(topString)}
                    </div>
                </div>`;
        }
    }

    // Handling the bottom placeholder
    if (!this._bottomPlaceholder) {
this._bottomPlaceholder =
this.context.placeholderProvider.tryCreateContent(
    PlaceholderName.Bottom,
    { onDispose: this._onDispose });

    // The extension should not assume that the expected placeholder is available.
    if (!this._bottomPlaceholder) {
        console.error('The expected placeholder (Bottom) was not found. ');
        return;
    }
}

```

```

}

if (this.properties) {
  let bottomString: string = this.properties.Bottom;
  if (!bottomString) {
    bottomString = '(Bottom property was not defined.)';
  }

  if (this._bottomPlaceholder.domElement) {
    this._bottomPlaceholder.domElement.innerHTML = `
      <div class="${styles.app}">
        <div class="ms-bgColor-themeDark ms-fontColor-white ${styles.bottom}">
          <i class="ms-Icon ms-Icon--Info" aria-hidden="true"></i> ${escape(bottomString)}
        </div>
      </div>`;
  }
}
}
}

private _onDispose(): void {
  console.log('[HelloWorldApplicationCustomizer._onDispose] Disposed custom top and bottom placeholders.');
```

```

?loadSPFX=true&debugManifestsFile=https://localhost:4321/temp/manifests.js&customActions={"81ba
a660-5d07-4666-9020-
5b9e65cda93d":{"location":"ClientSideExtension.ApplicationCustomizer","properties":{"Top":"Tekst v
glavi","Bottom":"Tekst v nogi"}}}
```

Pakiranje

Web scope / Tenant scope

gulp bundle

gulp package-solution

sharepoint/solution -> sppkg

List view command extensions

Location:

ClientSideExtension.ListViewCommandSet.ContextMenu

ClientSideExtension.ListViewCommandSet.CommandBar

ClientSideExtension.ListViewCommandSet