

MudBlazor Input Component (Picker)

Pada Blazor – Part 3

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Junindar, ST, MCPD, MOS, MCT, MVP

junindar@gmail.com

<http://junindar.blogspot.com>

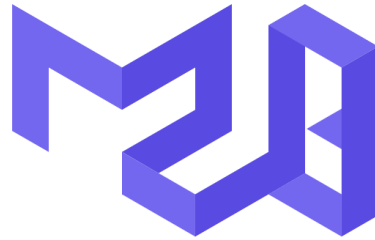
Abstrak

Blazor adalah Web Framework yang bersifat Open Source dimana aplikasi Web yang bersifat client-side interactive dapat dikembangkan dengan menggunakan .Net (C#) dan HTML. Pada saat ini C# biasa digunakan untuk melakukan proses back-end dari aplikasi web. Dengan menggunakan fitur baru dari ASP.NET Core yaitu Blazor, kita dapat membangun interactive WEB dengan menggunakan C# dan .NET .

Code .Net berjalan pada WebAssembly, yang artinya kita dapat menjalankan “NET“ didalam browser (Client) tanpa harus menginstall plugin seperti Silverlight, Java maupun Flash.

Pendahuluan

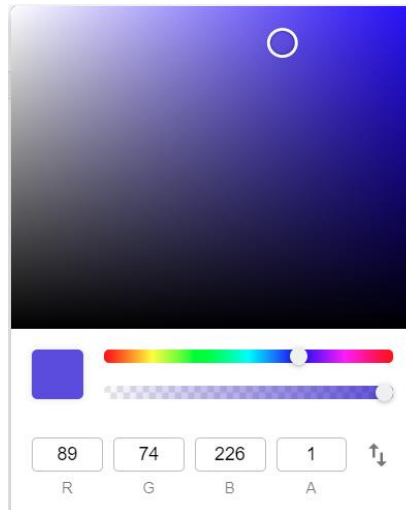
MudBlazor adalah sebuah material design component framework yang dibangun khusus untuk Blazor (<https://mudblazor.com/>). Terdapat banyak komponen pada MudBlazor seperti chart, grid dan lain-lain untuk membantu dalam membangun aplikasi web dengan menggunakan blazor. Seluruh komponen pada MudBlazor dibangun dengan menggunakan C# tanpa javascript kecuali jika sangat diperlukan. Dan dokumentasi untuk penggunaan MudBlazor ini sangat lengkap sehingga membantu para developer untuk menggunakannya.



Blazor Components For faster and easier web development

Artikel ini melanjutkan dari artikel sebelumnya, yang membahas komponen-komponen yang terdapat pada MudBlazor. Komponen-komponen tersebut sering digunakan dalam sebuah aplikasi web. Beberapa komponen yang akan dijelaskan pada artikel ini adalah Color Picker, Date Picker dan Time Picker. Dikarenakan artikel ini masih melanjutkan fitur-fitur yang ada pada MudBlazor, maka sebelum memulai latihan-latihan pada artikel ini, pastikan telah menyelesaikan latihan-latihan pada artikel sebelumnya yang dapat dilihat disini. <http://junindar.blogspot.com/2022/10/mudblazor-input-component-pada-blazor.html> dan <http://junindar.blogspot.com/2022/12/mudblazor-input-component-pada-blazor.html> .

- Color Picker



Color Picker adalah komponen yang digunakan untuk memilih warna dan untuk beberapa kasus digunakan untuk membuat skema warna.

Color Picker digunakan untuk memilih dan menyesuaikan nilai dari warna. Dalam desain grafis dan pengeditan gambar, pengguna biasanya memilih warna melalui antarmuka dengan representasi visual dari sebuah warna yang diatur dengan hue, saturation and lightness dimensions (HSL).

Desain grafis yang lebih berpengalaman sekalipun, biasanya membuat kesalahan saat harus memilih warna yang tepat untuk desain jenis apa pun.

Biasanya, nilai dari warna juga ditampilkan secara numerik, sehingga dapat diingat dengan tepat, seperti tiga nilai 0-255 yang masing-masing mewakili merah, hijau, dan biru.

Pada MudBlazor Color Picker dikenal dengan “MudColorPicker”.

1. Basic

```
<MudColorPicker Label="Basic Color Picker" @bind-Text="_colorValue"  
Style="@($"color: {_colorValue};")" Placeholder="Select Color" />  
  
@code {  
private string _colorValue;  
}
```

Untuk menggunakan Color Picker dapat dilihat pada sintaks diatas. Untuk mendapatkan nilai dari warna yang dipilih kita gunakan “@bind-Text” dan nilai tersebut ditampung pada field “_colorValue”. Selanjutnya kita dapat menggunakan property “style” untuk mengganti color pada text.

Basic

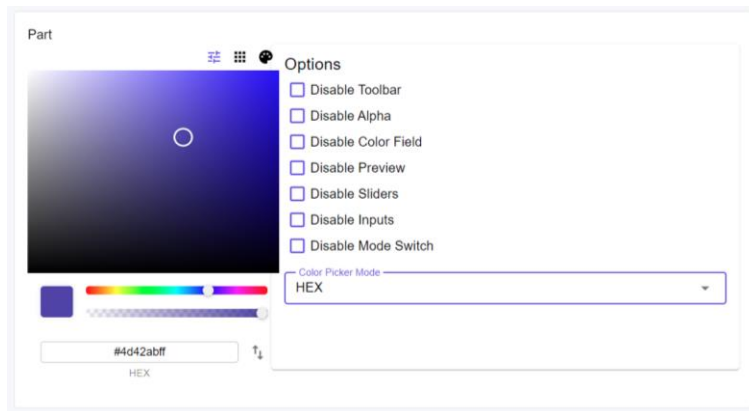
Basic Color Picker

#ad1515ff



2. Color Picker Fitur

Pada MudBlazor, Color Picker memiliki banyak fitur. Kita dapat mengaktifkan maupun menonaktifkan fitur-fitur tersebut sesuai dengan kebutuhan.

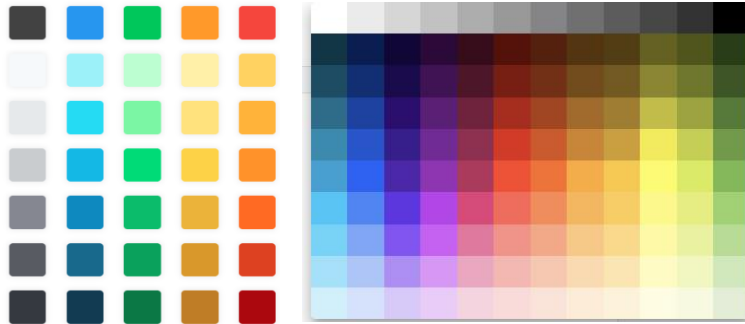


Dari gambar diatas dapat kita lihat berbagai macam fitur yang terdapat pada Color Picker, seperti Toolbar, Preview, slider dan lain-lain. Untuk mengaktifkan maupun menonaktifkan fitur-fitur tersebut kita dapat gunakan sintaks seperti dibawah.

```
<MudColorPicker DisableToolbar="DisableToolbar" DisableAlpha="DisableAlpha"
DisableColorField="DisableColorField" DisablePreview="DisablePreview"
DisableSliders="DisableSliders" DisableInputs="DisableInputs"
DisableModeSwitch="DisableModeSwitch"
ColorPickerMode="ColorPickerMode" PickerVariant="PickerVariant.Static" />
@code {
    public bool DisableToolbar { get; set; } = true;
    public bool DisableAlpha { get; set; }
    public bool DisableColorField { get; set; }
    public bool DisablePreview { get; set; }
    public bool DisableSliders { get; set; }
    public bool DisableInputs { get; set; }
    public bool DisableModeSwitch { get; set; }
    public ColorPickerMode ColorPickerMode { get; set; }
}
```

3. View

Terdapat beberapa View pada MudColorPicker. Tampilan default dapat diubah dengan menggunakan "ColorPickerView" dan dapat berguna disaat kita ingin menampilkan pilihan yang minimalis atau membatasi penggunaan warna-warna tertentu.



Terdapat 4 pilihan view pada MudColorPicker seperti pada sintaks dibawah.

```
<MudColorPicker Label="Spectrum" ColorPickerView="ColorPickerView.Spectrum" />  
<MudColorPicker Label="Palette" ColorPickerView="ColorPickerView.Palette" />  
<MudColorPicker Label="Grid" ColorPickerView="ColorPickerView.Grid" />  
<MudColorPicker Label="Compact Grid" ColorPickerView="ColorPickerView.GridCompact" />
```

4. Custom Palette

Kita juga dapat mengubah warna-warna yang akan ditampilkan pada Palette dengan menggunakan property Palette.



Sebelumnya kita buat terlebih dahulu daftar warna-warna yang ingin ditampilkan pada Color Picker, seperti dibawah ini.

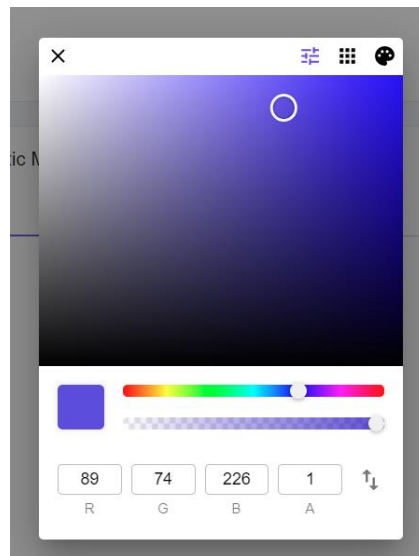
```
public IEnumerable<MudColor> CustomPalette { get; set; } = new MudColor[]  
{  
    "#F44336", "#E91E63", "#9C27B0", "#673AB7", "#3F51B5",  
    "#FFEB3B", "#FFCDD2", "#F8BBD0", "#E1BEE7", "#D1C4E9", "#C5CAE9",  
    "#F44336", "#F06292", "#BA68C8", "#9575CD", "#7986CB",  
    "#EF5350", "#EC407A", "#AB47BC", "#7E57C2", "#5C6BC0",  
    "#E53935", "#D81B60", "#8E24AA", "#5E35B1", "#3949AB"  
};
```

Lalu kita assign daftar warna diatas kedalam property Palette.

```
<MudColorPicker PickerVariant="PickerVariant.Static"  
ColorPickerView="ColorPickerView.Palette" Palette="CustomPalette" />
```

5. Variant

Terdapat 3 variant pada Color Picker yaitu Dialog, InLine dan Static. Variant Dialog akan membuat Color Picker menjadi *On Top* pada halaman web. Sedangkan InLine adalah default variant, dimana Color Picker akan muncul pada saat di klik, tanpa nge-block halaman web. Dan terakhir adalah Static Mode, Color Picker akan selalu muncul tanpa perlu pengguna nge-click color picker textbox. Untuk memilih variant yang akan ditampilkan, kita hanya perlu menggunakan property “*PickerVariant*”.



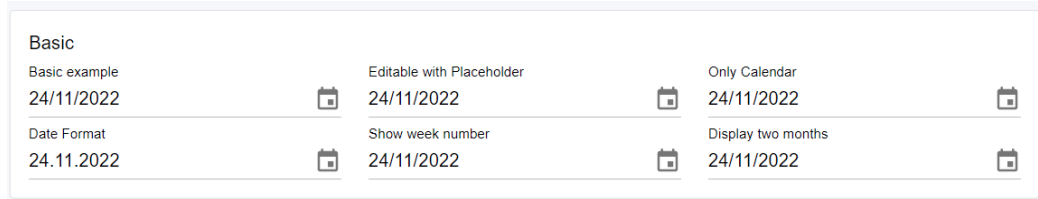
```
<MudColorPicker DisableToolbar="false" Label="Dialog Picker"
PickerVariant="PickerVariant.Dialog" />
<MudColorPicker DisableToolbar="false" Label="Dialog Picker"
PickerVariant="PickerVariant.Inline" />
<MudColorPicker PickerVariant="PickerVariant.Static" />
```

Diatas adalah contoh sintaks untuk mengganti variant pada MudColorPicker.

- Date Picker

Date Picker adalah komponen yang memungkinkan pengguna memilih tanggal dari kalender. Pada umumnya Date Picker menampilkan sebuah text box yang ketika diklik untuk memasukkan tanggal maka akan menampilkan kalender di samping atau di bawah text box, sehingga memungkinkan pengguna untuk memilih tanggal yang diinginkan. Dengan menggunakan Date Picker akan memudahkan dalam memvalidasi data, yaitu dengan membatasi rentang tanggal, seperti. hanya setelah hari ini dan dua minggu kemudian, atau hanya untuk tanggal di masa lalu.

1. Basic



Terdapat 6 Date Picker pada gambar diatas. Pada MudBlazor Date Picker dikenal dengan nama MudDatePicker. Untuk menampung nilai dari Date Picker kita perlu membuat sebuah field dengan tipe DateTime. Lalu gunakan property “@bind-Date” untuk melakukan *Bind* dengan field (date) yang telah kita buat sebelumnya.

```
<MudDatePicker Label="Basic example" @bind-Date="date"/>

<MudDatePicker Label="Editable with Placeholder" Editable="true" @bind-Date="date"
Placeholder="Select Date" />

<MudDatePicker Label="Only Calendar" @bind-Date="date" DisableToolbar="true" />

<MudDatePicker Label="Date Format" @bind-Date="date" DateFormat="dd.MM.yyyy" />

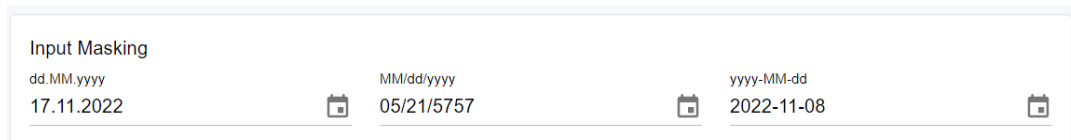
<MudDatePicker Label="Show week number" ShowWeekNumbers="true" @bind-Date="date" />

<MudDatePicker Label="Display two months" DisplayMonths="2"
TitleDateFormat="dddd, dd MMMM" @bind-Date="date" />

@code {
DateTime? date = DateTime.Today;
}
```

Pada Date Picker kedua, kita aktifkan textbox pada, sehingga memungkinkan pengguna untuk melakukan perubahan data langsung pada text box tanpa harus memilih dari kalender. Copy sintaks diatas dan amati perbedaan dari hasil yang didapat.

2. Input Masking



```
<MudDatePicker Label="dd.MM.yyyy" Editable="true" @bind-Date="date1" Mask="@(<new
DateMask("dd.MM.yyyy")>)" DateFormat="dd.MM.yyyy" Placeholder="de-AT Date" />

<MudDatePicker Label="MM/dd/yyyy" Editable="true" @bind-Date="date2" Mask="@(<new
DateMask("MM/dd/yyyy")>)" DateFormat="MM/dd/yyyy" Placeholder="en-US Date" />

<MudDatePicker Label="yyyy-MM-dd" Editable="true" @bind-Date="date3" Mask="@(<new
DateMask("0000-00-00")>)" DateFormat="yyyy-MM-dd" Placeholder="ISO Date" />
```

Dengan menggunakan property “Mask” kita dapat menentukan *string expression* untuk membatasi inputan yang akan dimasukkan oleh pengguna. Atau bisa kita sebut sebagai template atau format set yang harus sesuai dengan ekspresi yang telah ditentukan.

3. Culture

Pada MudDatePicker kita dapat mengganti nama bulan dan hari sesuai dengan Culture yang digunakan. Untuk melakukan hal tersebut kita gunakan property “Culture” dan masukkan spesifik Culture yang diinginkan (@CultureInfo.GetCultureInfo("zh-Hans")). Atau kita dapat membuat custom Culture Info sehingga nama bulan dan hari dapat kita ubah sesuai dengan yang kita inginkan.



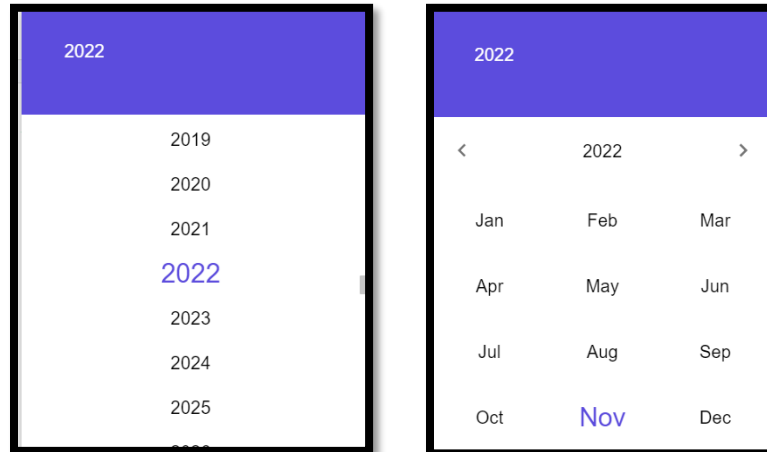
```
<MudDatePicker Label="Persian" @bind-Date="dateCulture"  
Culture="@GetPersianCulture()" UseShortNames="false" TitleDateFormat="dddd, dd MMMM"/>  
  
<MudDatePicker Label="Chinese" @bind-Date="dateCulture"  
Culture="@CultureInfo.GetCultureInfo("zh-Hans")" TitleDateFormat="dddd, dd MMMM" />
```

Untuk sintaks custom Culture Info dapat dilihat pada project lampiran.

4. View dan Variant

Default tampilan pada MudDatePicker pada saat dibuka adalah tanggal (date). Tapi kita dapat mengganti tampilan tersebut menjadi Year ataupun Month. Untuk melakukan hal tersebut, kita gunakan property “OpenTo” dan masukkan value OpenTo.Year atau Month, seperti pada sintaks dibawah.

```
<MudDatePicker Label="Year" OpenTo="OpenTo.Year" Text="2020-10-19" />  
<MudDatePicker Label="Month" OpenTo="OpenTo.Month" Text="2020-10-19" />
```

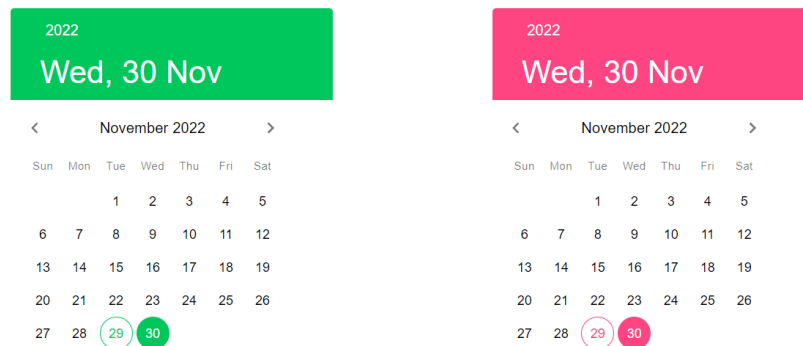



Seperti Color Picker, pada Date Picker di MudBlazor juga terdapat 3 variant yaitu Dialog, Inline dan Static. Property dan value yang digunakan juga sama yaitu “PickerVariant”.

```
<MudDatePicker PickerVariant="PickerVariant.Dialog" Label="Picker example"
Text="2020-10-19" />
```

5. Color dan Range

Agar warna dari kalender sesuai dengan halaman web site, maka kita dapat mengganti warna default dari kalender pada DatePicker. Cukup dengan menambahkan property “Color” dan masukkan nilai dari warna yang diinginkan.



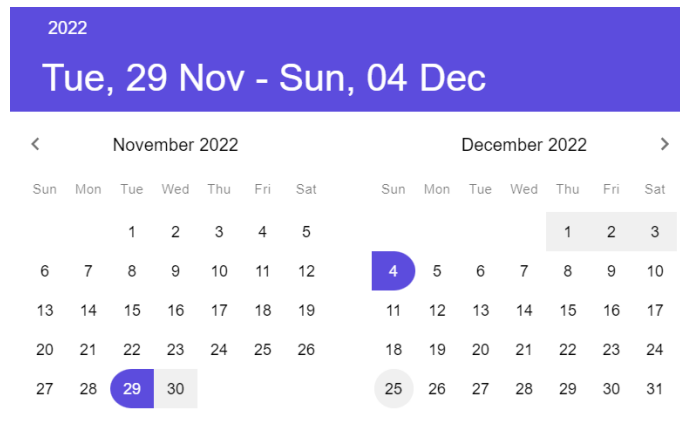
```
<MudDatePicker PickerVariant="PickerVariant.Static" Color="Color.Success"
Rounded="true" Date="@((DateTime.Today.AddDays(1))" />
```

```
<MudDatePicker PickerVariant="PickerVariant.Static" Color="Color.Secondary"
Rounded="true" Date="@((DateTime.Today.AddDays(1))" />
```

Terdapat fitur menarik pada MudDatePicker, dimana pada DatePicker bisa menggunakan 2 tanggal seperti tanggal awal dan tanggal akhir, atau biasa kita kenal dengan DateRange. Pada MudBlazor terdapat class "DateRange" yang dapat kita gunakan untuk fitur ini. Jika pada aplikasi memiliki field Start dan End Date, maka fitur ini akan sangat membantu. Dikarenakan kita cukup menggunakan satu DatePicker saja.

```
<MudDateRangePicker Label="Basic range picker" @bind-DateRange="_dateRange" />
```

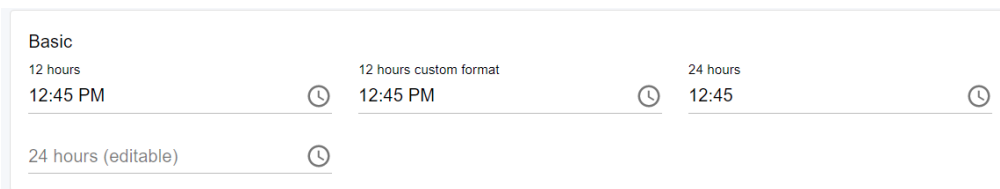
```
@code {  
    private DateRange _dateRange = new DateRange(DateTime.Now.Date,  
        DateTime.Now.AddDays(5).Date);  
}
```



- Time Picker

Time Picker adalah komponen yang mirip seperti Data Picker. Jika pada Date Picker pengguna dapat memilih tanggal pada kalender, sedangkan untuk Time Picker digunakan untuk memilih spesifik waktu.

1. Basic



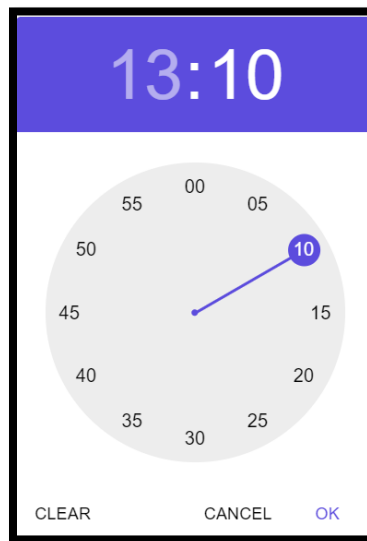
Terdapat 4 Time Picker pada gambar diatas. Pada MudBlazor Time Picker dikenal dengan nama MudTimePicker. Untuk menampung nilai dari Time Picker kita perlu membuat sebuah field dengan tipe TimeSpan. Lalu gunakan property

“@bind-Time“ untuk melakukan *Bind* dengan field (time) yang telah kita buat sebelumnya.

```
<MudTimePicker Label="12 hours" AmPm="true" @bind-Time="time" />  
  
<MudTimePicker Label="12 hours custom format" AmPm="true"  
TimeFormat="h:mm tt" @bind-Time="time" />  
  
<MudTimePicker Label="24 hours" @bind-Time="time" />  
  
<MudTimePicker Label="24 hours (editable)" Editable="true" />  
  
@code {  
    TimeSpan? time = new TimeSpan(00, 45, 00);  
}
```

Untuk menggunakan AM/PM format (12 jam), pastikan kita tambahkan property “AmPm” dan set value-nya menjadi true. Jika tidak maka Time Picker akan menggunakan format 24 jam. Selain itu kita juga bisa mengatur format dari waktu, dengan menggunakan property TimeFormat. Dan terakhir agar TimePicker bisa diubah nilainya kita harus mengaktifkan terlebih dahulu dengan menggunakan property Editable menjadi true.

2. Actions Button



Pada Time Picker, kita dapat menambahkan action button seperti pada gambar diatas. Untuk contoh diatas terdapat 3 buah action button yaitu Clear, Cancel dan OK. Button Clear berfungsi untuk menghapus nilai yang ada sebelumnya. Cancel untuk membatalkan pilihan dan menutup Time Picker dan terakhir adalah OK

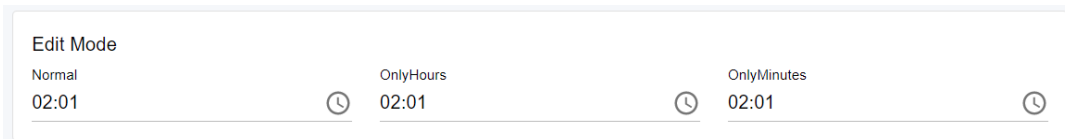
button dimana akan mengambil nilai yang telah dipilih. Pastikan untuk menonaktifkan AutoClose terlebih dahulu.

```
<MudTimePicker @ref="_picker" Label="With action buttons" AutoClose="false">
<PickerActions>
  <MudButton Class="mr-auto align-self-start"
    OnClick="@(() => _picker.Clear())">Clear</MudButton>
  <MudButton OnClick="@(() => _picker.Close(false))">Cancel</MudButton>
  <MudButton Color="Color.Primary"
    OnClick="@(() => _picker.Close())">Ok</MudButton>
</PickerActions>
</MudTimePicker>
@code {
  MudTimePicker _picker;
}
```

Untuk menggunakan Action Button kita tambahkan PickerActions yang didalamnya terdapat 3 buah MudButton.

3. Edit Mode

Dengan menggunakan property “TimeEditMode“ kita dapat membatasi pengeditan nilai pada waktu/jam hingga hanya memungkinkan mengubah nilai jam atau menit saja.



```
<MudTimePicker Label="Normal" @bind-Time="time"
TimeEditMode="TimeEditMode.Normal" />

<MudTimePicker Label="OnlyHours" @bind-Time="time"
TimeEditMode="TimeEditMode.OnlyHours" />

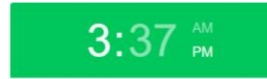
<MudTimePicker Label="OnlyMinutes" @bind-Time="time"
TimeEditMode="TimeEditMode.OnlyMinutes" />

@code {
  TimeSpan? time = new TimeSpan(00, 45, 00);
}
```

Terdapat 3 pilihan pada TimeEditMode yaitu Normal, OnlyHours dan OnlyMinutes.

4. Color

Seperti halnya pada Date Picker, Time Picker juga memiliki fitur untuk mengganti warna pada tampilannya. Property yang digunakan juga sama yaitu Color. Kita cukup memasukkan nilai warna apa yang kita inginkan pada property tersebut.



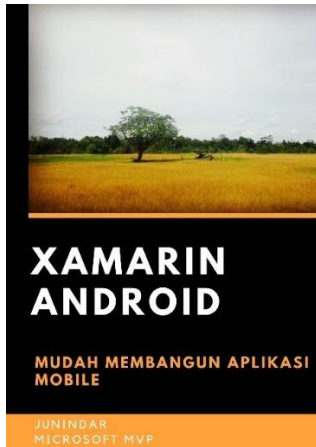
```
<MudTimePicker PickerVariant="PickerVariant.Static" Color="Color.Secondary"  
Rounded="true" Text="13:37"/>
```

Penutup

Sedangkan untuk memudahkan dalam memahami isi artikel, maka penulis juga menyertakan dengan full source code project latihan ini, dan dapat di download disini

<http://junindar.blogspot.com/2022/12/mudblazor-input-component-picker-pada.html>

Referensi



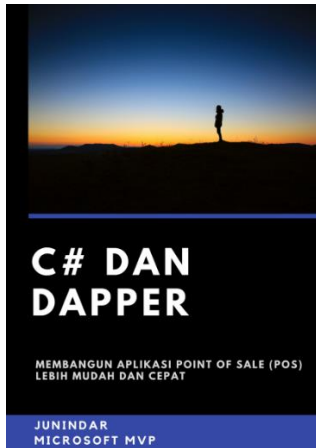
<https://play.google.com/store/books/details?id=G4tFDgAAQBAJ>



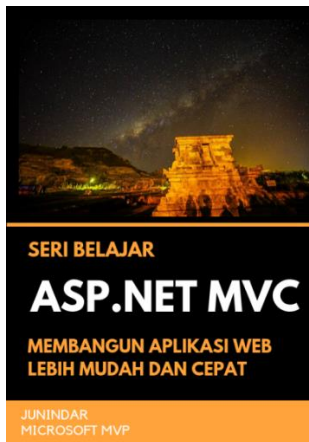
<https://play.google.com/store/books/details?id=VSLiDQAAQBAJ>



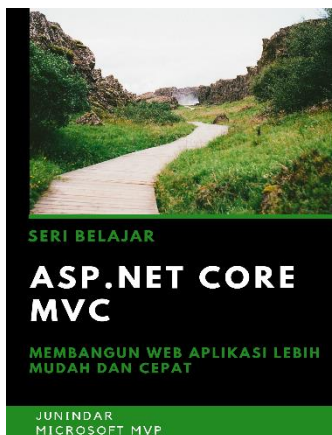
https://play.google.com/store/books/details/Junindar_Xamarin_Forms?id=6Wg-DwAAQBAJ



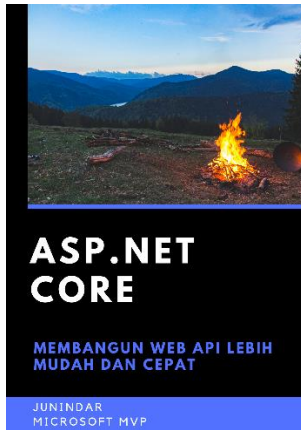
https://play.google.com/store/books/details/Junindar_C_dan_Dapper_Membangun_Aplikasi_POS_Point?id=6TErDwAAQBAJ



https://play.google.com/store/books/details/Junindar_ASP_NET_MVC_Membangun_Aplikasi_Web_Lebih?id=XLlyDwAAQBAJ



https://play.google.com/store/books/details/Junindar_ASP_NET_CORE_MVC?id=xEe5DwAAQBAJ



https://play.google.com/store/books/details/Junindar_ASP_NET_CORE?id=COUWEAAQBAJ



https://play.google.com/store/books/details/Junindar_Microsoft_Blazor_Membangun_Aplikasi_Web_D?id=HKZhEAAAQBAJ

Biografi Penulis.



Junindar Lahir di Tanjung Pinang, 21 Juni 1982. Menyelesaikan Program S1 pada jurusan Teknik Inscreenatika di Sekolah Tinggi Sains dan Teknologi Indonesia (ST-INTEN-Bandung). Junindar mendapatkan Award Microsoft MVP VB pertanggal 1 oktober 2009 hingga saat ini. Senang mengutak-atik computer yang berkaitan dengan bahasa pemrograman. Keahlian, sedikit mengerti beberapa bahasa pemrograman seperti : VB.Net, C#, SharePoint, ASP.NET, VBA. Reporting: Crystal Report dan Report Builder. Database: MS Access, MY SQL dan SQL Server. Simulation / Modeling Packages: Visio Enterprise, Rational Rose dan Power Designer. Dan senang bermain gitar, karena untuk bisa menjadi pemain gitar dan seorang programmer sama-sama membutuhkan seni. Pada saat ini bekerja di salah satu Perusahaan Consulting dan Project Management di Malaysia sebagai Senior Consultant. Memiliki beberapa sertifikasi dari Microsoft yaitu Microsoft Certified Professional Developer (MCPD – SharePoint 2010), MOS (Microsoft Office Specialist) dan MCT (Microsoft Certified Trainer) Mempunyai moto hidup: **“Jauh lebih baik menjadi Orang Bodoh yang giat belajar, dari pada orang Pintar yang tidak pernah mengimplementasikan ilmunya”**.