



Таймеры

Последнее обновление: 01.07.2016



- | | | | | |
|---|-------------------------------|---|--|--|
| 1 | Входные Двери в Минске | Входные двери от 574 рублей для квартиры или дома. Установка и доставка. Бесплатный замер. maxmidminsk.by | | |
| 2 | Попробуй SE Ranking | Проверка позиций сайта от 1 коп, анализ конкурентов и SEO аудит сайта. 14 дней бесплатно! seranking.ru | | |
| 3 | Необычный подарок | Интерьерные 3D-скульптуры из дизайнерского картона. Легко собирать! paperraz.by | | |

Применить таймер в Xamarin Forms можно разными способами. Первый способ представляет использование метода **Device.StartTimer()**, который определен в пространстве имен Xamarin.Forms. Он имеет следующее определение:

```
Device.StartTimer(TimeSpan interval, Func<bool> callback);
```

Первый параметр задает временный интервал, через который будет действовать метод из второго параметра - функция обратного вызова. Причем эта функция должна возвращать булево значение: true - если надо, чтобы таймер продолжал выполнение, и false, если надо, чтобы таймер остановился. Например:

```
using System;
using Xamarin.Forms;

namespace HelloApp
{
    public partial class MainPage : ContentPage
    {
        Button timerButton;
        bool alive = true;
        public MainPage()
        {
            timerButton = new Button
            {
                VerticalOptions = LayoutOptions.Center,
                HorizontalOptions = LayoutOptions.Center,
                FontSize = Device.GetNamedSize(NamedSize.Large, typeof(Button))
            };
            timerButton.Clicked += TimerButton_Clicked;

            Content = timerButton;

            Device.StartTimer(TimeSpan.FromSeconds(1), OnTimerTick);
        }
        private bool OnTimerTick()
        {
            timerButton.Text = DateTime.Now.ToString("T");
            return alive;
        }
        private void TimerButton_Clicked(object sender, EventArgs e)
        {
            if(alive==true)
            {
                alive = false;
            }
            else
            {
                alive = true;
                Device.StartTimer(TimeSpan.FromSeconds(1), OnTimerTick);
            }
        }
    }
}
```

Здесь определена кнопка, на которую выводится текущее время. Для вывода времени запущен таймер, причем он срабатывает раз в секунду (TimeSpan.FromSeconds(1)). По таймеру срабатывает метод OnTimerTick, который возвращает

значение переменной `alive`. Через нажатие на кнопку мы можем проконтролировать значение этой переменной и тем самым остановить или снова запустить таймер.



Еще одну форму таймера представляет симуляция таймера с помощью временной задержки **Task.Delay()**:

```
using System;
using System.Threading.Tasks;
using Xamarin.Forms;

namespace HelloApp
{
    public partial class MainPage : ContentPage
    {
        Button timerButton;
        bool alive = true;
        public MainPage()
        {
            timerButton = new Button
            {
                VerticalOptions = LayoutOptions.Center,
                HorizontalOptions = LayoutOptions.Center,
                FontSize = Device.GetNamedSize(NamedSize.Large, typeof(Button))
            };
            timerButton.Clicked += TimerButton_Clicked;

            Content = timerButton;

            DisplayTime();
        }
        private async void DisplayTime()
        {
            while (alive)
            {
                timerButton.Text = DateTime.Now.ToString("T");
                await Task.Delay(1000);
            }
        }
        private void TimerButton_Clicked(object sender, EventArgs e)
        {
            if(alive==true)
            {
                alive = false;
            }
            else
            {
                alive = true;
                DisplayTime();
            }
        }
    }
}
```

```

    }
}
}

```

Этот практически аналогичный пример за тем исключением, что тут явным образом запускается асинхронный метод `DisplayTime()`, который обрабатывает во вторичном потоке. Поэтому, несмотря на то, что он запускается в конструкторе, после завершения конструктора он продолжит работу.

Освойте вёрстку сейчас

Сверстайте ваш первый сайт под руководством опытного наставника. htmlacademy.ru



[Назад](#) [Содержание](#) [Вперед](#)



Сетка "Рабица", ворота, калитки, столбы

рабица.бел



Яндекс.Директ

Гибкий кабель-канал

promtechnolog.ru



Яндекс.Директ

9 Комментариев

metanit.com

1 Войти

Рекомендовать

Поделиться

Лучшее в начале



Присоединиться к обсуждению...

ВОЙТИ С ПОМОЩЬЮ

ИЛИ ЧЕРЕЗ DISQUS ?

Имя



Sasha Belov • 11 дней назад

Здравствуйте. Пробовал запускать таймер с `ViewModel`. Зделал задержку в 45 сек, затем имела изменять значение поля. Но он не работал, в чем возможна ошибка?

```

public void StartTimer() {
    Device.StartTimer(new TimeSpan(0, 0, 45), () => {
        TimeLeft++;
        return false;
    });
}

```

Посоветуйте еще какие-то способы реализовать таймер с `ViewModel`. Спасибо.

^ | v • Ответить • Поделиться ›



Balalamb • месяц назад

Добрый день!

Подскажите, как правильно вызывать `Device.StartTimer`, если он должен вызывать асинхронный метод.

^ | v • Ответить • Поделиться ›

**Phenman** → Balalamb • 17 дней назад

```
private void EntryPointMethod()
{
    Device.StartTimer(TimeSpan.FromMilliseconds(intVvalue), () =>
    {
        Task.Factory.StartNew(async () =>
        {
            //полезная работа
            EntryPointMethod(); // если надо запускать таймер циклично
        });
        return false;
    });
}
```

1 ^ | ▾ • Ответить • Поделиться ›

**Balalamb** → Phenman • 4 дня назад

Спасибо!

^ | ▾ • Ответить • Поделиться ›

**AZinX** • 5 месяцев назад

Подскажите в чем проблема?

Есть страница, на ней ListView с ItemTapped="schlsList_ItemTapped" в xaml.

Есть обработчик этого события в коде страницы:

```
private void schlsList_ItemTapped(object sender, ItemTappedEventArgs e)
{
    var vm = this.BindingContext as SchedulesListViewModel;
    vm.ItemTapped();
}
```

который вызывает метод из ListViewModel:

```
private bool DoubleTapped { get; set; }
```

```
public void ItemTapped()
{
```

```
    DoubleTapped = true;
```

```
    Device.StartTimer(TimeSpan.FromMilliseconds(500), () => { DoubleTapped = false; return false; });
```

[показать больше](#)

^ | ▾ • Ответить • Поделиться ›

**AZinX** → AZinX • 5 месяцев назад

Решил проблему :)

Кому интересно, то сделал так:

В коде страницы определил два поля:

```
private int count = 0;
private bool _switcher = true;
```

Обработчик нажатия кнопки:

```
private void schlsList_ItemTapped(object sender, ItemTappedEventArgs e)
{
    count += 1;
    _switcher = true;
```

```
    Device.StartTimer(TimeSpan.FromMilliseconds(500), () => { _switcher = false; count = 0; return false; });
```

```
    if (_switcher == true && count == 2)
    {
        count = 0;
```

[показать больше](#)[^](#) | [v](#) • [Ответить](#) • [Поделиться](#) ›**Василь Шагала** • 9 месяцев назад

Подскажите как сделать таймер обратного отсчета к примеру на 1 минуту Xamarin.Forms
Зарание спасибо

[^](#) | [v](#) • [Ответить](#) • [Поделиться](#) ›**Mikhail Ukhin** → Василий Шагала • 9 месяцев назад

```
// start = TimeSpan.FromMinutes(1);
Device.StartTimer(TimeSpan.FromSeconds(1), () =>
{
    start -= TimeSpan.FromSeconds(1);
    return start != TimeSpan.Zero;
});
```

[^](#) | [v](#) • [Ответить](#) • [Поделиться](#) ›**sdadsadas dsadasd** • год назад

Подскажите, почему у меня не работают таймеры. Они просто игнорируются.

Пример:

```
editor1.Text = "1";
Device.StartTimer (new TimeSpan (0, 0, 3), () => {
    return false;
});
editor1.Text = "2";
Device.StartTimer (new TimeSpan (0, 0, 3), () => {
    return false;
});
editor1.Text = "3";
```

По логике каждые три секунды текст должен меняться. А выходит так, что текст сразу же становится "3".

[^](#) | [v](#) • [Ответить](#) • [Поделиться](#) ›

ТАКОЖЕ НА METANIT.COM

Go | Соответствие интерфейсу

1 комментарий • 16 дней назад



Пу — Еще давай. Не останавливайся, пожалуйста. Это гениальный язык. Он, конечно, не божественен, как шарп, но гениальный.

Vue.js | Компоненты в отдельных файлах и Webpack

2 комментария • 3 месяца назад



Metanit — package.json"scripts": { "dev": "webpack-dev-server --hot --open", "build": "webpack" },

Angular в ASP.NET Core | Маршрутизация

2 комментария • 2 месяца назад



Zaur YakubOff — Если создал проект на основе Asp.NetCore MVC куда нужно вставить <base href="/" /> ???

Vue.js | Валидация props

1 комментарий • 4 месяца назад



eugene81 — Подключена минимизированная версия библиотеки - сообщения об ошибках не показываются. При подключении не сжатой - видны

Подписаться Добавить Disqus на свой сайт [Добавить Disqus](#) [Добавить](#) Конфиденциальность