NFC Questions - a text based game for Windows Phone

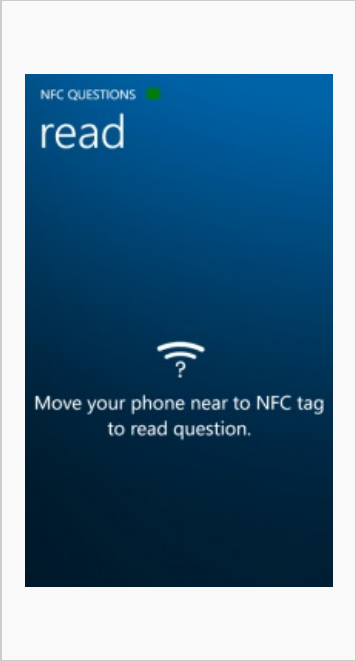
This article explains how to use Near Field Communication (NFC) to create text based "game" in Windows Phone 8.

10 Nov 2013

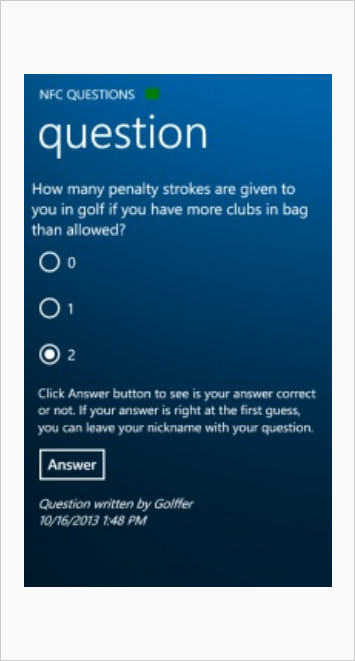
Introduction

This code example shows how to use NFC on Windows Phone to read and write text based questions and answers from/to NFC tag. This application can be used like a game - the first player can save a question and answers to the NFC tag; following players can write their own questions to NFC tag if they know the right answer.

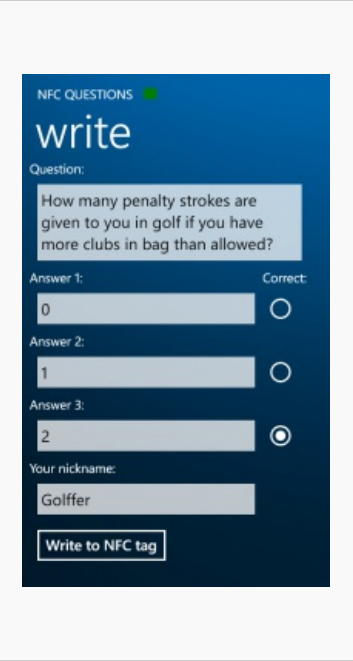
The application contains following pages: **Read**, **Question** and **Write**.



Read



Question



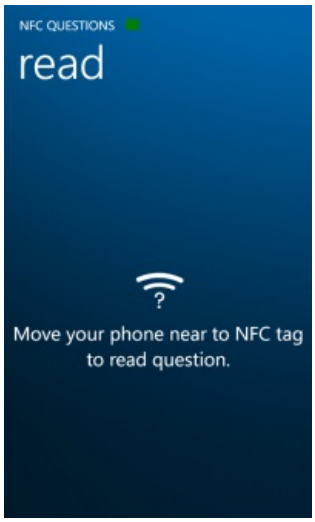
Write

Note: You can learn the basics of NFC from Use NFC tags with Windows Phone 8.

Read question from the NFC tag

Layout (MainPage.xaml)

The application first page contains one `StackPanel` which includes one `Image` and one `TextBlock` (`InfoTextBlock`). This `InfoTextBlock` prompts app users to move phone near to NFC tag. This same `TextBlock` will be used to show other information too.



(C) Copyright Nokia 2014. All rights reserved.

```
<Grid x:Name="ContentPanel" Grid.Row="1" Margin="12,0,12,0">
            <StackPanel VerticalAlignment="Center" HorizontalAlignment="Center">
            <Image Grid.Row="0" HorizontalAlignment="Center" VerticalAlignment="Top"
Height="100" Width="100" Source="/Assets/icon.png"/>
            <TextBlock Grid.Row="1"
                       x:Name="InfoTextBlock"
                       TextWrapping="Wrap"
                       Text="Move your phone near to NFC tag to read question."
                       VerticalAlignment="Center" HorizontalAlignment="Center"
FontSize="30"
                       TextAlignment="Center"
                       Foreground="White"
                       />
            </StackPanel>
        </Grid>
```

Programming (MainPage.xaml.cs)

In `OnNavigatedTo` method we first check that the end user device supports NFC and initialize device arrived and device departed callback methods. This example stores question to NFC tag with `PublishBinaryMessage` method using `Windows:WriteTag.NFCQuestions` as a protocol and subtype. We try to read this same information from NFC tag using `SubscribeForMessage` method later.

```
protected override void OnNavigatedTo(NavigationEventArgs e)
{
 base.OnNavigatedTo(e);

 InfoTextBlock.Text = "Move your phone near to NFC tag to read question.";
 NfcTagIndicator.Background = new System.Windows.Media.SolidColorBrush(Colors.Red);
 // check if device has NFC
 device = ProximityDevice.GetDefault();
 if (device != null)
 {
  timerDisposed = false;
  questionFound = false;
  timerStarted = false;
  device.DeviceArrived += device_DeviceArrived;
  device.DeviceDeparted += device_DeviceDeparted;
  questionSubscribeId = device.SubscribeForMessage("Windows.NFCQuestions",
NFCQuestionsHandler);
 }
 else
 {
  MessageBox.Show("Your phone has no NFC or NFC is disabled");
  return;
 }
}
```

The `device_DeviceArrived` method will be called when the phone is near to the NFC tag. First, the information text on the screen will be changed, so the end user knows that NFC tag will be read.

The application uses timer behind to check/wait if there a question available or not. The timer will wait 2 seconds and if `NFCQuestionHandler` is not called in that time, then the first question can be written to NFC tag.

```
// NFC tag is near
```


(C) Copyright Nokia 2014. All rights reserved.

```
private void device_DeviceArrived(ProximityDevice sender)
{
 Dispatcher.BeginInvoke(() =>
 {
  InfoTextBlock.Text = "NFC Tag found, read question please wait...";
  NfcTagIndicator.Background = new System.Windows.Media.SolidColorBrush(Colors.Green);
  if (!questionFound && !timerStarted)
  {
   timer = new Timer(MyTimerCallback, InfoTextBlock, 2000, Timeout.Infinite);
   timerStarted = true;
  }
 });
}
```

The `MyTimerCallback` method will be called after 2 seconds when the NFC tag is available and empty NFC tag is used. First the timer will be disposed, application is not waiting for message from the NFC tag anymore and application navigates to the "write question" page.

```
private void MyTimerCallback(object state)
{
   Dispatcher.BeginInvoke(() =>
   {
    // dispose timer, there is no question in tag
    if (!timerDisposed)
    {
     timer.Dispose();
     timerDisposed = true;
    }
    // stop subscribing to read question
    device.StopSubscribingForMessage(questionSubscribeId);
    MessageBox.Show("There is no question in the NFC tag. You can now set the first
question.");
    NavigationService.Navigate(new Uri("/WritePage.xaml?answersCount=-1",
UriKind.Relative));
   });
}
```

The `NFCQuestionHandler` method will be called if there is a question available in NFC tag. First timer will be disposed (if it still running) and because question is found we will stop subscribing to read it. Question and answer are stored in one string and *Regex* will be used to parse questions and answers from one string. The application will navigate to the Answer page if all is ok.

```
// get question data from NFC tag
private void NFCQuestionsHandler(ProximityDevice sender, ProximityMessage message)
{
 questionFound = true;
 // release timer (question found)
 if (timerStarted && !timerDisposed)
 {
  timer.Dispose();
  timerDisposed = true;
 }
 // stop subscribing to read question
 device.StopSubscribingForMessage(questionSubscribeId);
 // get question and answers
 string question = message.DataAsString;
```


(C) Copyright Nokia 2014. All rights reserved.

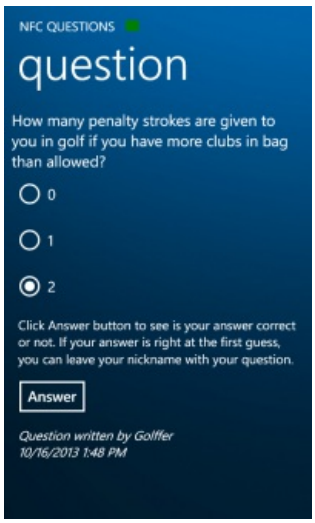
```
 // store to strings
 string[] strings = Regex.Split(question, delimeter);
 // check that there is que,del,ans1,del,ans2,del,ans3,del,rightans
 if (strings.Count() < 13)
 {
  InfoTextBlock.Text = "Cannot read question and answers from NFC Tag.";
  return;
 }
 // got question data -> got to answer page
 Dispatcher.BeginInvoke(() =>
 {
    NavigationService.Navigate(new Uri("/AnswerPage.xaml?q=" + question,
UriKind.Relative));
 });
}
```

Answer the question

Layout (AnswerPage.xaml)

Question page is designed to support all the screen sizes in Windows Phone 8. You can read more about Multi-resolution apps for Windows Phone 8 here.

This page contains 7 rows: `TextBlock`, three `RadioButton`s, `TextBlock`, `Button` and one `TextBlock`. Each of these rows height is set to auto and no margins are used.




```
<Grid x:Name="ContentPanel" Grid.Row="1" Margin="12,0,12,0">
 <Grid.RowDefinitions>
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
 </Grid.RowDefinitions>
 <TextBlock Grid.Row="0" x:Name="QuestionTextBlock" HorizontalAlignment="Left"
TextWrapping="Wrap"
      VerticalAlignment="Top" Width="Auto" Height="Auto"
      Text="Question text" FontSize="24" Foreground="White"/>
 <RadioButton Grid.Row="1" x:Name="Answer1RadioButton" Content="Answer 1"
HorizontalAlignment="Left" VerticalAlignment="Top" Foreground="White"/>
```


(C) Copyright Nokia 2014. All rights reserved.

```
 <RadioButton Grid.Row="2" x:Name="Answer2RadioButton" Content="Answer 2"
HorizontalAlignment="Left" VerticalAlignment="Top" Foreground="White" />
 <RadioButton Grid.Row="3" x:Name="Answer3RadioButton" Content="Answer 3"
HorizontalAlignment="Left" VerticalAlignment="Top" Foreground="White" />
 <TextBlock Margin="10" Grid.Row="4" HorizontalAlignment="Left" TextWrapping="Wrap"
Text="Click Answer button to see is your answer correct or not. If your answer is right
at the first guess, you can leave your nickname with your question."
VerticalAlignment="Top" Foreground="White"/>
 <Button Grid.Row="5" x:Name="AnswerButton" Content="Answer" HorizontalAlignment="Left"
VerticalAlignment="Top" Click="AnswerButton_Click" Foreground="White"/>
 <TextBlock Margin="10" Grid.Row="6" x:Name="InfoTextBlock" HorizontalAlignment="Left"
TextWrapping="Wrap" Text="TextBlock" VerticalAlignment="Top" FontStyle="Italic"
Foreground="White"/>
</Grid>
```

Programming (AnswerPage.xaml.cs)

`OnNavigatedTo` method will parse question and answers to the screen. Question will be passed here from the `MainPage` as a parameter.

```
// page is navigated to
protected override void OnNavigatedTo(NavigationEventArgs e)
{
 base.OnNavigatedTo(e);

 string question = "";
 if (NavigationContext.QueryString.TryGetValue("q", out question))
 {
  string[] strings = Regex.Split(question, delimeter);
  QuestionTextBlock.Text = strings[0];
  Answer1RadioButton.Content = strings[2];
  Answer2RadioButton.Content = strings[4];
  Answer3RadioButton.Content = strings[6];
  correctAnswer = strings[8];
  InfoTextBlock.Text = "Question written by " + strings[10] + Environment.NewLine
+strings[12];
 }
}
```

`AnswerButton_Click` method will be called when the user wants to answer the question. Application will navigate to `WritePage` if the answer is correct. Answer count will be passed to `WritePage`. Users can store nickname with the question if the question is answered correctly in the first guess.

```
private void AnswerButton_Click(object sender, RoutedEventArgs e)
{
 bool correct = false;
 answersCount++;

 if (Answer1RadioButton.IsChecked == true && correctAnswer == "1")
 {
  correct = true;
 }
 else if (Answer2RadioButton.IsChecked == true && correctAnswer == "2")
 {
  correct = true;
```


(C) Copyright Nokia 2014. All rights reserved.

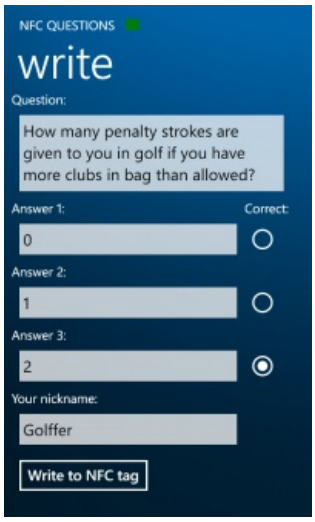
```
 }
 else if (Answer3RadioButton.IsChecked == true && correctAnswer == "3")
 {
  correct = true;
 }
 else
 {
  correct = false;
 }

 if (correct)
 {
  MessageBoxResult m = MessageBox.Show("You answer is correct, do you want to insert
your own question to NFC tag now?", "Correct answer", MessageBoxButton.OKCancel);
  if (m == MessageBoxResult.OK)
  {
   NavigationService.Navigate(new Uri("/WritePage.xaml?answersCount="+answersCount,
UriKind.Relative));
  }
 }
 else
 {
  MessageBox.Show("Sorry, but your answer is incorrect.");
 }
}
```

Write a new question

Layout (WritePage.xaml)

Write question page is designed to support all the screen sizes in Windows Phone 8 (like all of this application pages). There is eleven rows used. Each element has height property used, so Auto value is used in `RowDefinitions` heights. Few controls are positioned in the same row. This can be done using `ColumnDefinitons` and set second element `Grid.Column` to 1. The Question `TextBox` takes both columns of space. This can be done by setting `Grid.ColumnSpan` value to 2.




```
<Grid x:Name="ContentPanel" Margin="12,133,12,0" Grid.RowSpan="2">
 <Grid.RowDefinitions>
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
```


(C) Copyright Nokia 2014. All rights reserved.

```
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
  <RowDefinition Height="Auto" />
 </Grid.RowDefinitions>
 <Grid.ColumnDefinitions>
  <ColumnDefinition Width="Auto" />
  <ColumnDefinition Width="Auto" />
 </Grid.ColumnDefinitions>
 <TextBlock Grid.Row="0" HorizontalAlignment="Left" TextWrapping="Wrap" Text="Question:"
VerticalAlignment="Top"/>
 <TextBox Grid.Row="1" Grid.ColumnSpan="2" MaxLength="200" x:Name="QuestionTextBox"
HorizontalAlignment="Left" Height="143" TextWrapping="Wrap" VerticalAlignment="Top"
Width="436"/>
 <TextBlock Grid.Row="2" HorizontalAlignment="Left" TextWrapping="Wrap" Text="Answer 1:"
VerticalAlignment="Top" Width="187"/>
 <TextBlock Grid.Row="2" Grid.Column="1" HorizontalAlignment="Right" TextWrapping="Wrap"
Text="Correct:" VerticalAlignment="Top" Width="83"/>
 <TextBox Grid.Row="3" MaxLength="50" x:Name="Answer1TextBox" HorizontalAlignment="Left"
Height="72" TextWrapping="Wrap" VerticalAlignment="Top" Width="363"
RenderTransformOrigin="0.479,0.153"/>
 <RadioButton Grid.Row="3" Grid.Column="1" IsChecked="True" x:Name="A1RadioButton"
Content="" HorizontalAlignment="Left" VerticalAlignment="Top"
RenderTransformOrigin="0.426,0.194"/>
 <TextBlock Grid.Row="4" HorizontalAlignment="Left" TextWrapping="Wrap" Text="Answer 2:"
VerticalAlignment="Top" Width="187"/>
 <TextBox Grid.Row="5" MaxLength="50" x:Name="Answer2TextBox" HorizontalAlignment="Left"
Height="72" TextWrapping="Wrap" VerticalAlignment="Top" Width="363"/>
 <RadioButton Grid.Row="5" Grid.Column="1" x:Name="A2RadioButton" Content=""
HorizontalAlignment="Left" VerticalAlignment="Top" RenderTransformOrigin="0.574,0.097"/>
 <TextBlock Grid.Row="6" HorizontalAlignment="Left" TextWrapping="Wrap" Text="Answer 3:"
VerticalAlignment="Top" Width="187"/>
 <TextBox Grid.Row="7" MaxLength="50" x:Name="Answer3TextBox" HorizontalAlignment="Left"
Height="72" TextWrapping="Wrap" VerticalAlignment="Top" Width="363"/>
 <RadioButton Grid.Row="7" Grid.Column="1" x:Name="A3RadioButton" Content=""
HorizontalAlignment="Left" VerticalAlignment="Top"/>
 <TextBlock Grid.Row="8" HorizontalAlignment="Left" TextWrapping="Wrap" Text="Your
nickname:" VerticalAlignment="Top" Width="187"/>
 <TextBox Grid.Row="9" MaxLength="50" x:Name="NameTextBox" HorizontalAlignment="Left"
Height="72" TextWrapping="Wrap" VerticalAlignment="Top" Width="363"/>
 <Button Grid.Row="10" x:Name="WriteButton" Content="Write to NFC tag"
HorizontalAlignment="Left" VerticalAlignment="Top" Click="WriteButton_Click"/>
</Grid>
```

Programming (WritePage.xaml.cs)

`OnNavigatedTo` method and `OnNavigatedFrom` uses State collection to save and read `TextBox` values if this page goes to back (so end users is not loosing the dynamic texts). Look more information from the source codes.

`WriteButton_Click` method will be called when the end user wants to write question and answers to the NFC tag (question, answers and nickname cannot be empty). `DataWriter` class is used to save text in UTF-8 format. `Windows:WriteTag.NFCQuestions` protocol is used save content to NFC tag.

(C) Copyright Nokia 2014. All rights reserved.

```
private void WriteButton_Click(object sender, RoutedEventArgs e)
{
 if (QuestionTextBox.Text.Length == 0)
 {
  MessageBox.Show("Question is empty!");
  return;
 }
 if (Answer1TextBox.Text.Length == 0)
 {
  MessageBox.Show("Answer one is empty!");
  return;
 }
 if (Answer2TextBox.Text.Length == 0)
 {
  MessageBox.Show("Answer two is empty!");
  return;
 }
 if (Answer3TextBox.Text.Length == 0)
 {
  MessageBox.Show("Answer three is empty!");
  return;
 }
 if (NameTextBox.Text.Length == 0)
 {
  MessageBox.Show("Name is empty!");
  return;
 }
 // check correct answer
 string correct = "";
 if (A1RadioButton.IsChecked == true) correct = "1";
 else if (A2RadioButton.IsChecked == true) correct = "2";
 else if (A3RadioButton.IsChecked == true) correct = "3";

 // write question
 string date = DateTime.Now.ToShortDateString() + " " +
DateTime.Now.ToShortTimeString();
 var dataWriter = new DataWriter() { UnicodeEncoding = UnicodeEncoding.Utf8 };
 dataWriter.WriteString(QuestionTextBox.Text + "xPTMx" +
         Answer1TextBox.Text + "xPTMx" +
         Answer2TextBox.Text + "xPTMx" +
         Answer3TextBox.Text + "xPTMx" +
         correct + "xPTMx" +
         NameTextBox.Text + "xPTMx" +
         date
         );
 publishId = device.PublishBinaryMessage("Windows:WriteTag.NFCQuestions",
dataWriter.DetachBuffer(), QuestionTransmitted);
 WriteButton.IsEnabled = false;
}
```

`QuestionTransmitted` method will be called after data is written to NFC tag. Here we first stop publishing question and navigate back to the first page of the application (question can be read again to test it works).

```
private void QuestionTransmitted(ProximityDevice sender, long messageId)
{
 // stop publishing question
```


(C) Copyright Nokia 2014. All rights reserved.

```
 device.StopPublishingMessage(publishId);
 Deployment.Current.Dispatcher.BeginInvoke(() =>
 {
  MessageBox.Show("Question, answers and your nickname are now written to the NFC tag.
Application first page will be shown again.");
  if (!fromMain) NavigationService.RemoveBackEntry(); // remove answer page if it exists
  NavigationService.GoBack();
 });
}
```

Summary

This code example shows how to use NFC in Windows Phone 8 (and later). I hope you find this article useful and it helps you work with NFC in your Windows Phone apps.

You can download source code from here: File:PTMNFCQuestions.zip

(C) Copyright Nokia 2014. All rights reserved.