**现代操作系统应用开发实验报告**

**学号：** 15331046 **班级 ：** 软工一班

**姓名：** 陈志扬 **实验名称：** HW5-Todos

1. **参考资料**

请在这里列出对本实验有帮助你所参考的资料或者网站。

参考资料：老师上课所用课件

参考网站：[创建自适应磁贴](https://docs.microsoft.com/zh-cn/windows/uwp/controls-and-patterns/tiles-and-notifications-create-adaptive-tiles)里面的相关内容（本地磁贴等）；[借助本地通知使用通知队列](https://msdn.microsoft.com/library/windows/apps/xaml/hh868234)；

[发送磁贴更新](https://msdn.microsoft.com/zh-cn/library/windows/apps/xaml/hh868253)；[共享数据](https://docs.microsoft.com/zh-cn/windows/uwp/app-to-app/share-data)等

1. **实验步骤**

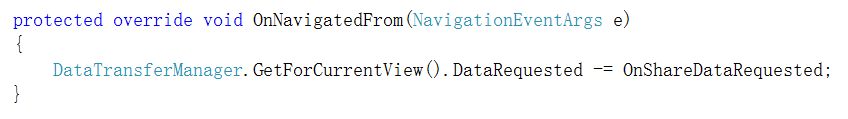
请在这里简要写下你的实验过程。

1. 在本次实验中，我在HW3的作业上来修改，即是说HW3作为HW5的demo。
2. 由于一开始我觉得磁贴的内容可能得花费一些时间来学习，所以我先做了共享数据这一块。共享数据因为不要求共享动态图片，所以也比较简单，只需给定图片路径然后利用一些函数接口即可实现。具体实现过程在第四部分详述。

注册事件：



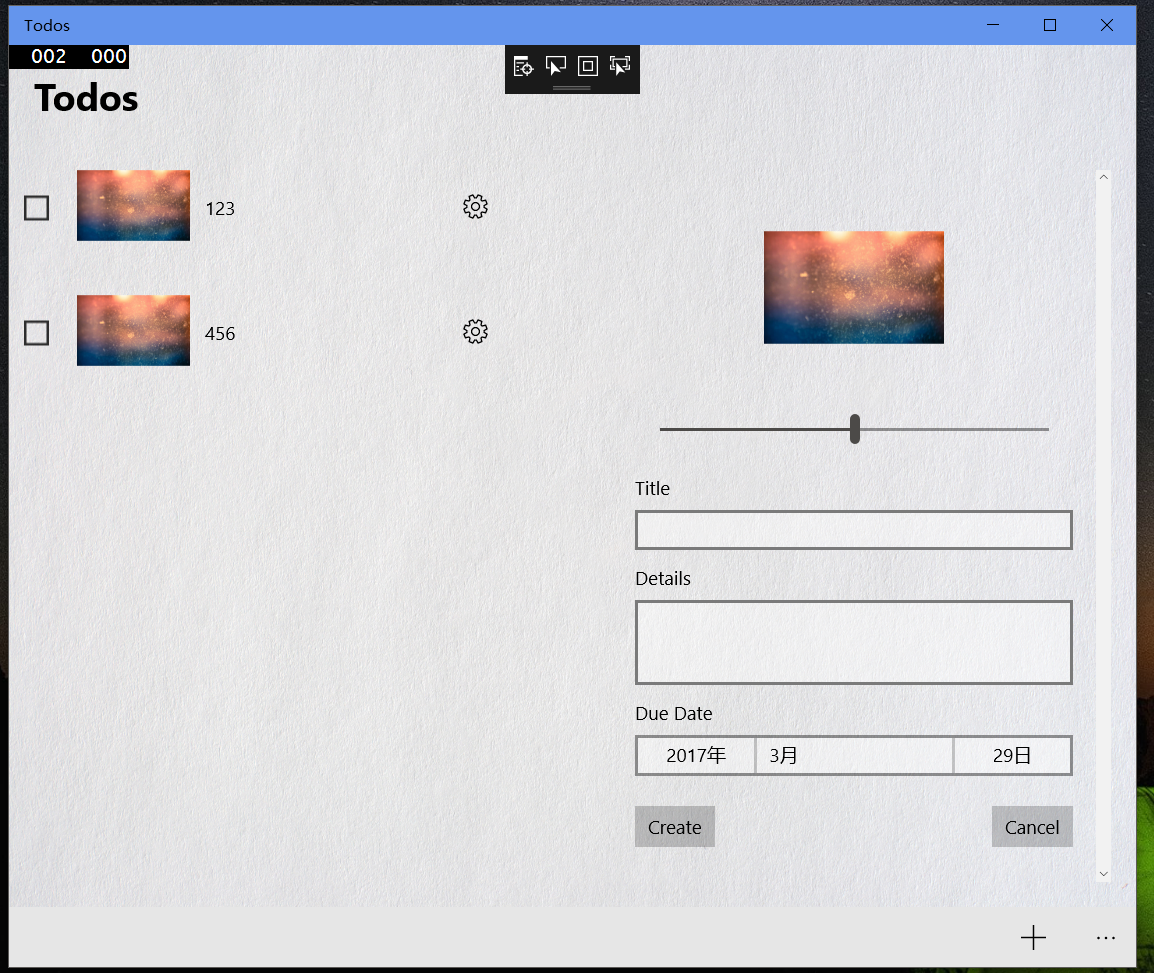
处理share后删除事件：



1. 然后是磁贴的实现。根据TA和相关网站的指导，首先安装了Notifications Visualize，根据里面的模板学习xml，然后自己在项目文件根目录中新建文件tile.xml，编写相关代码。
2. 接着，在MainPage.xaml.cs和NewPage.xaml.cs中编写xml加载函数updateTile，可用于解析xml，实现磁贴本地通知。
3. 最后，完善相关功能。例如，创建新的Todo item，磁贴能够更新，并循环显示。
4. **实验结果截图**

请在这里把实验所得的运行结果截图。

这是我添加的StoreLogo，启动界面



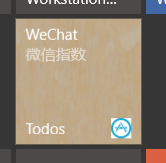
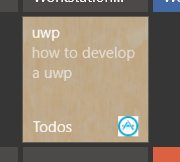
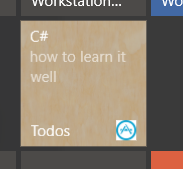
固定到开始屏幕，可看到如下的磁贴：



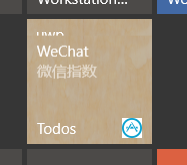
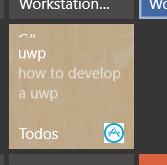
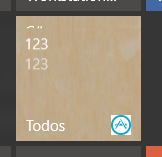
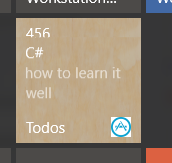
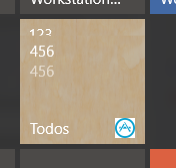
默认的只有小、中、宽三种大小，为其添加了“大”的尺寸大小



下面是创建的五个Todo Items，按以下顺序排列

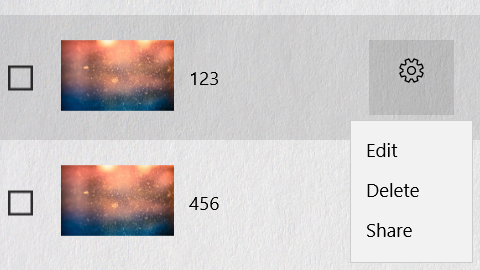


这里是体现轮询过程，五个Items按队列顺序依次展现在磁贴中：（截图比较难弄，但至少还是可以看出来变化状态的）

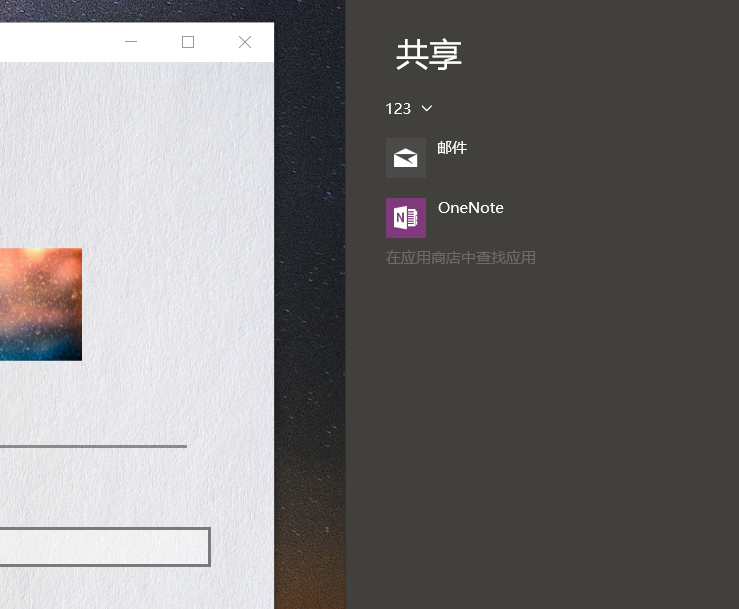


下面是共享数据的截图：

点击Share：



在屏幕右侧弹出共享目标：由此也可见共享标题正确。

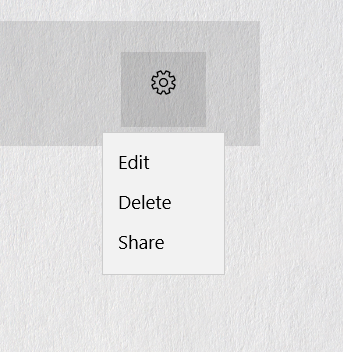


点击邮件应用作为共享目标，可见标题、文本内容、图片均有，说明已基本实现所要求的功能。

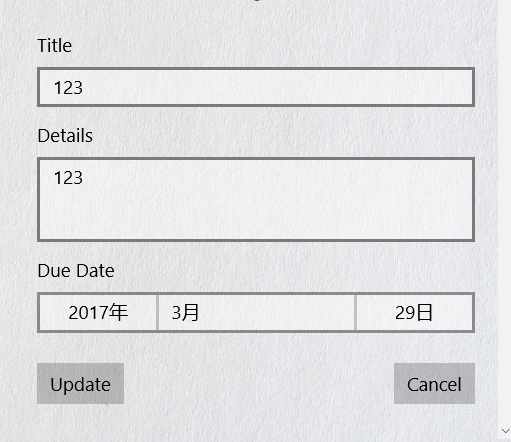


下面是我所做的补充功能，点击Setting按钮，可实现Edit，Delete功能，其中Edit对应Update功能(UpdateButton\_Clicked)，Delete对应直接删除功能(DeleteButton\_Clicked)

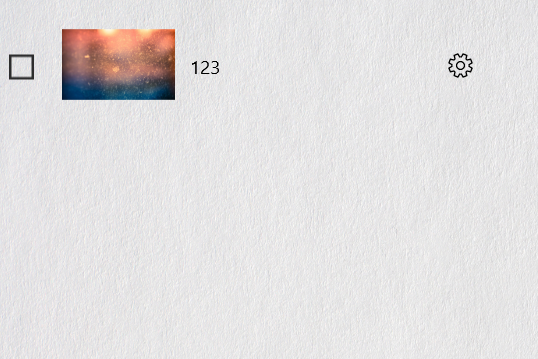
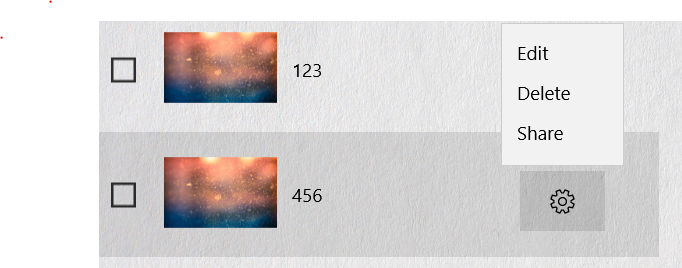
点击Edit：



宽屏下直接在右边修改，窄屏下跳转到NewPage修改，实现过程和HW3类似。



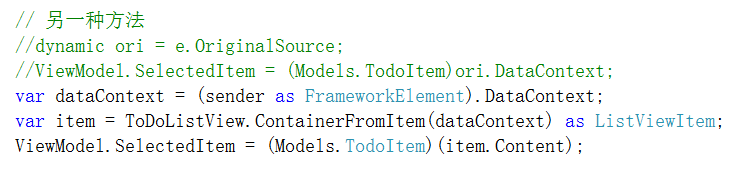
点击Delete：直接删除掉对应的Todo Item。



1. **实验过程遇到的问题**

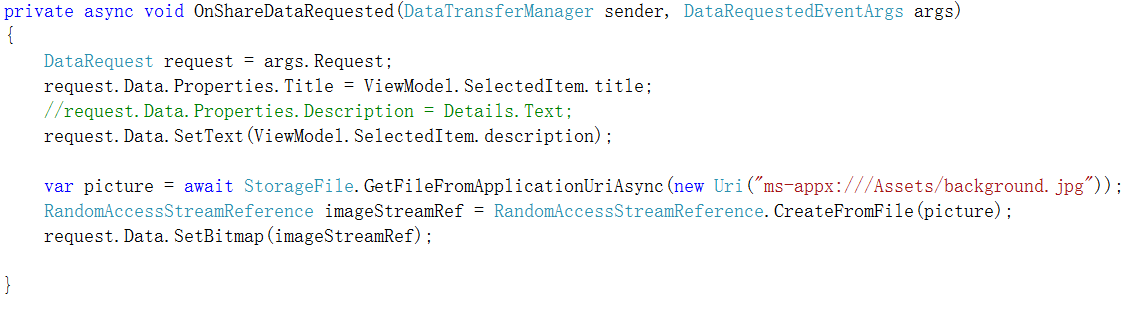
请在这里写下你在实验过程中遇到的问题以及解决方案。

* 1. 点击Share，不知道如何得到该Todo Item的内容，以为之前的Name.text依然可以使用，但其实是不行的。其实需要获取FrameworkElement参与数据绑定时的数据上下文，将其转化为ListViewItem。下面的代码可解决这个问题。

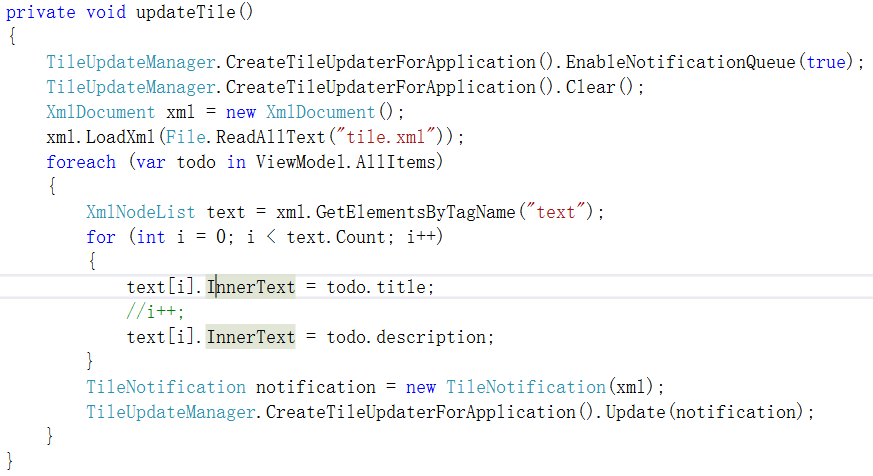


* 1. 将本地图片作为共享数据时读取图片无效。可以从文件类StorageFile中的一个方法

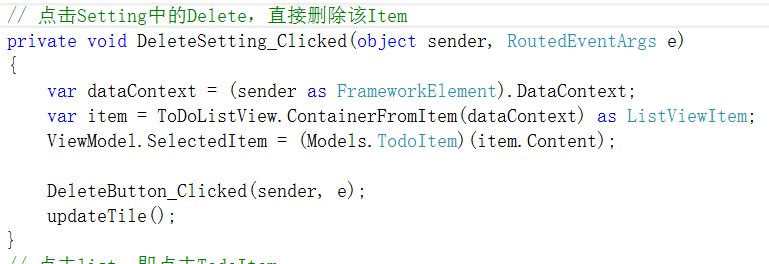
GetFileFromApplicationUriAsync(Uri uri)获得文件流，这里我漏了ms-appx导致一直出错。然后以输入和输出流的形式提供对数据流的随机访问，最后调用DataPackage.SetBitmap方法即可。具体代码如下：



* 1. 动态磁贴的加载问题。找了很多文档，知道了LoadXml但是一直无效。仔细看了LoadXml的参数，原来是string类型，改成LoadXml(File.ReadAllText(“tile.xml”))即可。
  2. 接下来是使用通知队列。这个参考文档给的很详细，主要是在这个实验我写了个foreach的循环，将每个Todo Item的title、description数据逐一赋值到tile.xml文件中。具体代码如下：



* 1. 说说针对Setting按钮补充的一些功能。主要是点击Edit对应Update，可以实现更新，点击Delete可以直接删除该Item，然后再调用updateTile函数实现磁贴更新。具体代码如下：



1. **思考与总结**

请在这里写下你本次试验的心得体会以及所思所想。

这次实验主要是学习自适应磁贴和共享数据，主要学到下面几个函数接口的操作等等。

TileUpdateManager.CreateTileUpdaterForApplication().EnableNotificationQueue(true);

TileNotification notification = new TileNotification(xml);

TileUpdateManager.CreateTileUpdaterForApplication().Update(notification);

XMLDocument.LoadXml(File.ReadAllText("tile.xml"));

DataTransferManager.GetForCurrentView().DataRequested

DataTransferManager.ShowShareUI();

TA说这次比较简单，就不给demo了，我只想说如果只是简单实现基本功能，确实不是很难，但是考虑到大磁贴显示多条item我还是没能解决问题，再者，如果共享要考虑到图片是绑定的图片，那难度又上升了。对Setting按钮的补充功能中，Edit对应update虽然可以修改，但是不知道为什么磁贴更新不了（确切地说应该是时好时坏）。学到现在，现操的代码真的感觉迷，接下来还有期中项目的uwp，真的要好好努力，付出更多精力来学习。

1. 实验报告提交格式为pdf。
2. 实验内容不允许抄袭，我们要进行代码相似度对比。如发现抄袭，按0分处理。