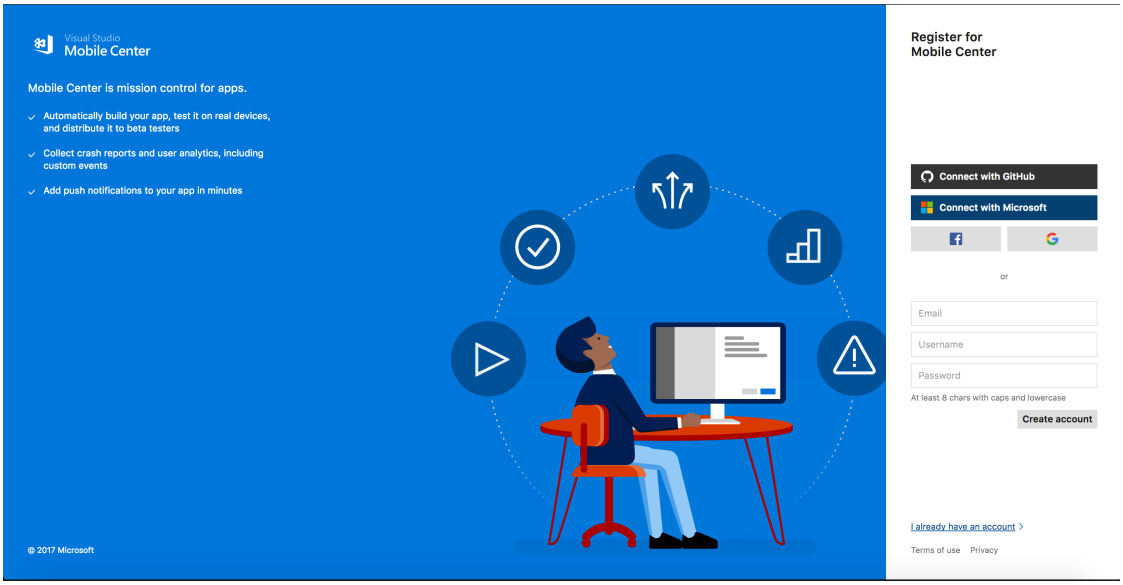
Microsoft

Ignite 2017 Pre-Day Training
Lab Manual: Visual Studio Mobile Center

Copyright © 2017 Microsoft, All Rights Reserved

Create a Mobile Center account

To get started with Mobile Center, you first need to create a free Mobile Center account. You can register with your existing Microsoft account, GitHub account, Google Account or Facebook account. Or you can create a new Mobile Center account with an email, username, and password.

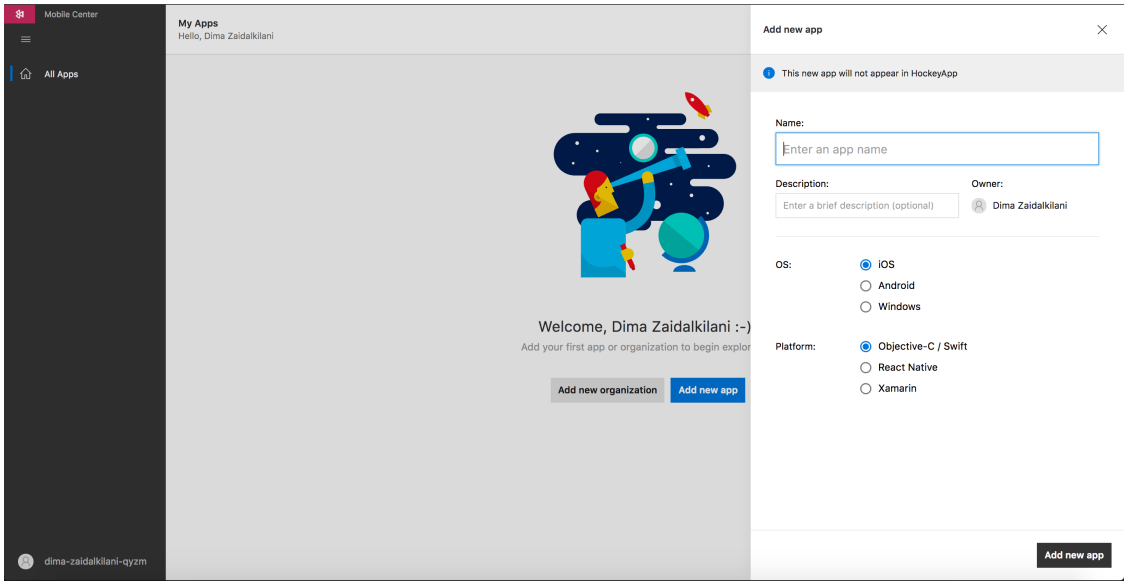


Copyright © 2017 Microsoft, All Rights Reserved

Create a Mobile Center app

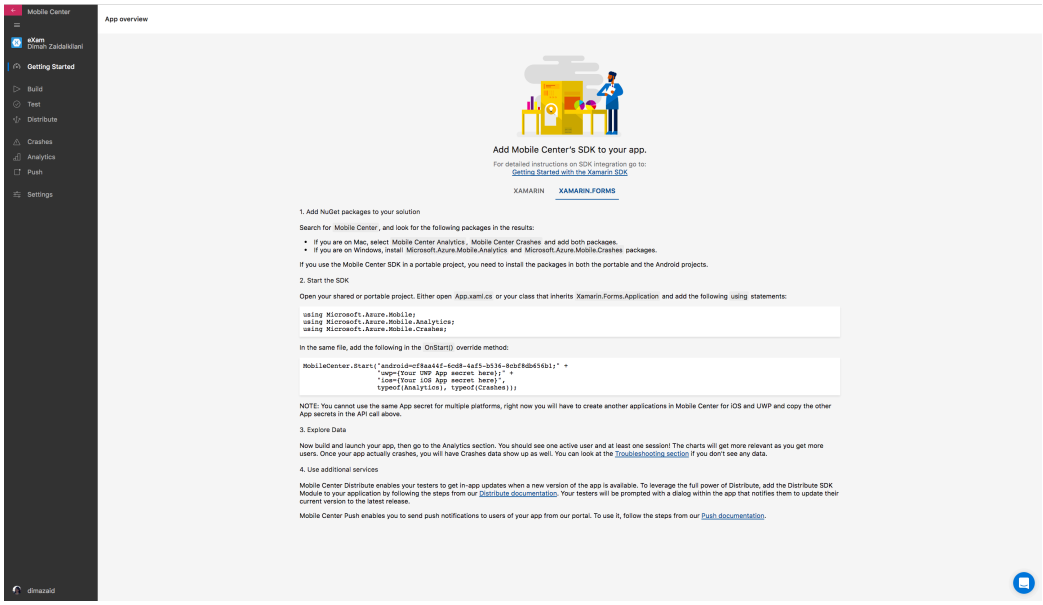
Now that you have created a Mobile Center account, you will enter the new account experience.

Click on "Add a new app" button in the middle of the screen if this is your first time accessing Mobile Center. Otherwise the "Add New" button will show up on the top right side of the screen. Choose "New App". Below is the new app form:



First, you need to give your app a name. You can choose any name, though to make it easier to identify, we recommend making the app name in Mobile Center match the app's actual name.
Next, choose the owner of the app. The owner can be a user (by default it's you), or it can be an org.
Finally, choose your app's OS and platform. In our example we're creating a Xamarin Android app, so you will choose "Android" for the OS, and "Xamarin" for the platform. Please note that if you are building a cross-platform codebase into multiple platforms (for example, building Xamarin.Forms iOS and Android apps), you will create a Mobile Center app for each platform.
Click on "Add new app", and within few seconds, you should see the appropriate Getting Started page for your selected app type.

Copyright © 2017 Microsoft, All Rights Reserved

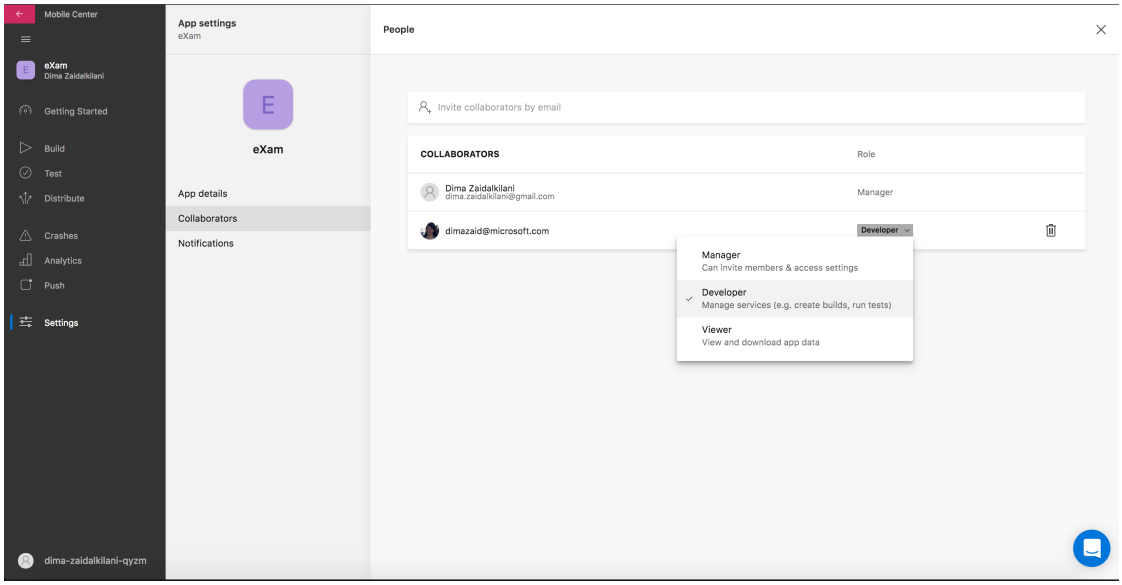


You can revisit your app's settings in the future by going to Settings.
Another way to create an app is by creating a Mobile Center organization first and then creating your apps inside that organization. If you know that your team will be collaborating on more than one app, you should create an organization and add your apps to it. For example, if you plan to build the iOS, Android, and UWP versions of the eXam app, it's recommended to keep them in the same org. More about Mobile Center organization can be found here.

Copyright © 2017 Microsoft, All Rights Reserved

Add collaborators to your app

To share your app with others, select an app and then click **Settings->People** to add collaborators by typing in the user's email address.



On each app there are three roles:

- Managers can manage app settings, collaborators, and integrations.
- Developers can manage app services (e.g. create builds, run tests).
- Viewers can view and download all data but cannot make changes.

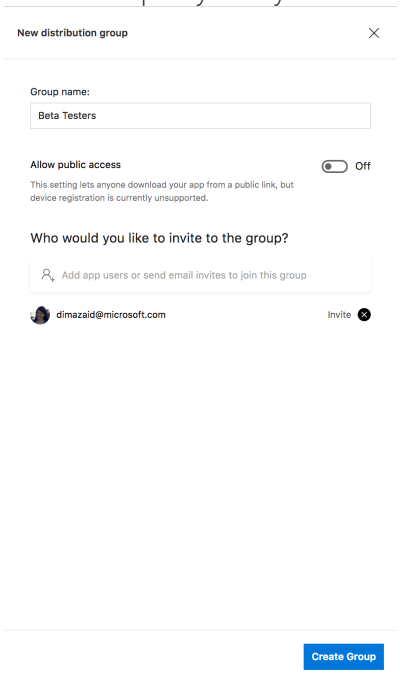
For every app you create, whether it be owned by you or an organization, you automatically become a 'Manager'.

Copyright © 2017 Microsoft, All Rights Reserved

Distribution

Mobile Center Distribution is a tool to for developers to release application binaries to their end user's devices.
In this step will create a "**Distribution Group**" which we will use in a later step to distribute your app's binaries.

1. Go to your app and click on "**Distribute**".
2. Click on "**New Group**"
3. Give the group a name, for example "Beta testers" and add at least one user to the list. For simplicity add your own user ID.

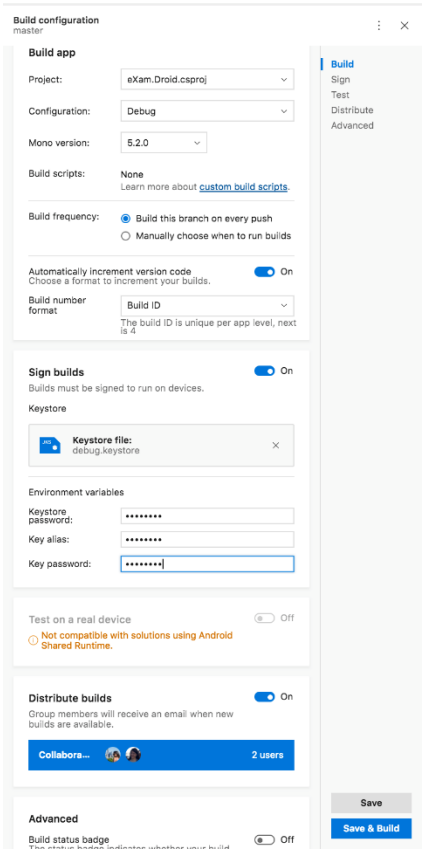


Copyright © 2017 Microsoft, All Rights Reserved

Build integration

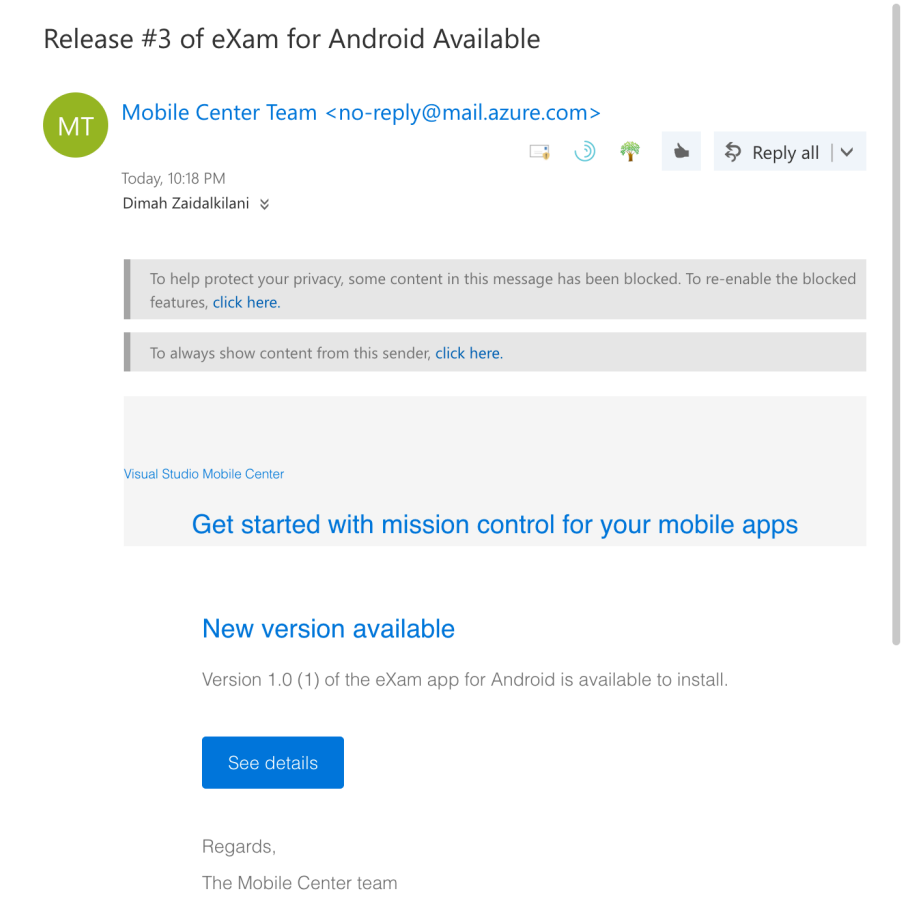
In this part, you will set up a CI system that sends the output of your build to the distribution group so they can receive the latest version of the application.

1. Upload the "eXam" project you created this morning to a Github repo
2. Go to Mobile Center and click on "**Build"**
3. Click on "Github" from the list. If you did not create your account with your Github ID, you may have to log into GitHub again at this point.
4. Choose your repo that you wish to connect to the Mobile Center Build service
5. Click on the master branch
6. Click on "Configure build"
7. Turn on "Automatically increment version code"
8. Turn on Sign builds. This will require you to upload the app's Keystore file. Follow this link to learn how to create your app's certificate.
   Note: This step is required to be able to distribute your app at the end of the build.
9. Turn on Distribute builds and choose the Distribution group you've created earlier
10. Click on "Save and build"



Copyright © 2017 Microsoft, All Rights Reserved

11. Once the build succeeds you and everyone in your will get an email that informs you about the new app/version of the app and a link to download it from your phone.



Copyright © 2017 Microsoft, All Rights Reserved

Microsoft

Crashes and Analytics getting started

In this part we will get our app ready to start collecting both Analytics and Crashes from your app.

1. Add the Mobile Center Analytics SDK to your shared project, and the Android project:

   - Visual Studio for Mac or Xamarin Studio
     Under your project, select Packages, open context menu and click Add packages.
     Search for Mobile Center, and select Mobile Center Analytics and Mobile Center Crashes.
     Click Add Packages.

   - Visual Studio for Windows
     Navigate to the Project > Manage NuGet Packages...
     Search for Mobile Center, then install Microsoft.Azure.Mobile.Analytics and Microsoft.Azure.Mobile.Crashes packages.

   - Package Manager Console
     Make sure the Package Manager Console is opened in either Xamarin Studio or Visual Studio. You will have to install an add-in for Xamarin Studio.
     Type the following commands:
     PM> Install-Package Microsoft.Azure.Mobile.Analytics PM> Install-Package Microsoft.Azure.Mobile.Crashes

2. Add the using statements:
   Open your App.cs and add the following lines below the existing using statements:

```
using Microsoft.Azure.Mobile;
using Microsoft.Azure.Mobile.Analytics;
using Microsoft.Azure.Mobile.Crashes;
```


Copyright © 2017 Microsoft, All Rights Reserved

3. Add the Start() method
   Open the Mobile Center's website, go to your app. Click on the "Getting Started" and copy the following:

```
MobileCenter.Start("android=[Your app's secret]",
                   typeof(Analytics), typeof(Crashes));"
```

4. Open App.cs and paste the "MobileCenterStart()" call inside the OnStart() method.

Copyright © 2017 Microsoft, All Rights Reserved

Microsoft

Analytics

Mobile Center Analytics is a mobile apps measurement tool that lets developers understand their end-user population and usage patterns.
In this section, you will start sending custom events every time a user starts an eXam and at the end of the exam along with the user's score.

1. Start tracking events
   Inside the same method "OnStart()" track "Exam Started" event:

```
Analytics.TrackEvent("Exam Started");
```

2. Open file "ReviewPageViewModel.cs" and add the following inside "Result Display" function:

```
Analytics.TrackEvent("Completed Exam", new
Dictionary<string, string> {{"Score", ""+CorrectCount } });
```

3. Go to Analytics and click on "Log flow" to see the events as they're being sent from your app.
4. Click on Overview to see the reports and verify you have sent the data

Copyright © 2017 Microsoft, All Rights Reserved

Crashes

In the previous section we added the Crashes SDK and started using it in the app, now let's put it to test!

1. In the "QuestionPage.xaml" add the following:

```
<Button
    x:Name="btnCrashme"
    AutomationId="Crashme"
    Grid.Row="5"
    Grid.Column="1"
    Text="Crash Me!"
    TextColor="White"
    BackgroundColor="#08D046"
    Command="{Binding CrashMe}">
</Button>
```

2. In "QuestionPageViewModel.xaml", inside class "QuestionPageViewModel" add the following:

```
public Command CrashMe { get; set; }
```

3. Inside QuestionPageViewModel (Game game)

```
CrashMe = new Command(OnCrash);
```

4. Add a new function in the same class;

```
Public void OnCrash(){
Crashes.GenerateTestCrash();
}
```

5. Run your app and click on the "Crash Me" button you just added.
6. Restart your app
7. Go to Crashes and explore the Crash report

Copyright © 2017 Microsoft, All Rights Reserved

8. You can checkout the **Events** to see the events this user was doing before the app crashed.

Copyright © 2017 Microsoft, All Rights Reserved

Test

Mobile Center Test is a UI test automation service for native and hybrid mobile apps. Tests written using supported frameworks can be run with little modification on hundreds of unique device model and operating system configurations hosted in a Microsoft data center. We will use the same Xamarin.Forms app we created earlier, and use Xamarin.UITest as the test framework.

To get the Mobile Center Test working against your application follow these steps:

1. Install the Mobile Center CLI
2. Go to the Mobile Center -> Test service and click on create a new test run.
3. Choose your devices that you would like to test against.
4. Select the Test series and your test framework. In this example, we're using Xamarin.UITests
5. Follow the steps in the Test Wizard output and copy the following command from the directory that contains the NuGet Package.

```
mobile-center test run uitest --app "<APP NAME>" --devices
"<DEVICE SET NAME>" --app-path <PATH_TO_APK> --test-series
"<TEST SERIES>" --locale "en_US" --build-dir <PATH TO UITEST
BUILD DIRECTORY>
```

6. Once you've uploaded your test, go back to Mobile Center->Test and click on the test run.
7. Explore the test results and note the CPU and Memory along with every UI test.

Copyright © 2017 Microsoft, All Rights Reserved