В октябре 2015 года вышел в general availability сервис Azure Mobile Engagement.

**Какую проблему он решает?**

Мы написали мобильное приложение, опубликовали его, пользователь его скачал, и больше мы о своём детище ничего не знаем. Что он делает, когда и как использует? Мы можем написать сами сервис по сбору аналитики с приложения (время/деньги) или использовать уже готовый сервис. Аналитика - это хорошо, но, имея знания, нужно уметь их использовать, нужно иметь канал связи с пользователем. Можно написать свою систему нотификаций (время на имплементацию-> деньги)

**Azure Mobile Engagement** <https://azure.microsoft.com/en-us/documentation/services/mobile-engagement/> комбинирует в себе эти 2 функции: сбор данных о поведении пользователя и сегментация пользователей и канал обратной связи с ним, причем для 3 популярных платформ (ios, android, windows).

Самое классное - яркое объяснение в виде 1.5-минутного видео <https://azure.microsoft.com/en-us/documentation/videos/azure-mobile-engagement-overview/>

“Бизнес-Пример”: мы написали игру, но пользователь со временем стал в нее реже играть… a значит, скоро может вообще уйти… и точно не будет покупать ништяки для своего персонажа. Для любой игры нужна постоянная вовлеченность пользователя. Mobile Engagement позволяет выделить сегмент игроков, готовых уйти, и вы можете начислить этому игроку какую-нибудь плюшку (премиумный танк, к примеру, как ему начислить - это уже имплементация на стороне вашего приложения) и отправить нотификацию, в которой поздравить игрока с этой плюшкой. Возможно, пользователь вновь загорится игрой.

Или более простой пример: игрок дошел до предпоследнего уровня игры. Не сложно представить, что скоро он пройдет игру и с большой вероятностью перестанет играть. Вы можете предложить ему прислать нотификацию, что “только для вас, лучшая цена на продолжение культовой саги”.

С вопросом «зачем нужен этот сервис?» мы закончили. Переходим к имплементации.

Для использования этого сервиса нужно совмещать 2 роли: разработчика и аналитика.

Введем несколько концепций <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-concepts/>:

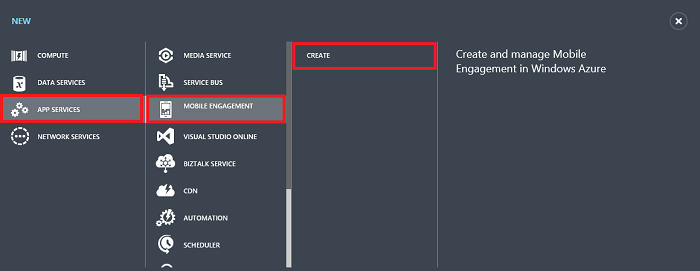
* User- тут ничего интересно.
* Session- набор активностей от входа в приложение до его закрытия.
* Activity- некоторой действие, совершенное за время сессии.
* Event- тип activity, у которого не было длительности (нажатие на кнопку).
* Job- тип активности, у которой есть начальный и конечный момент времени (http call).

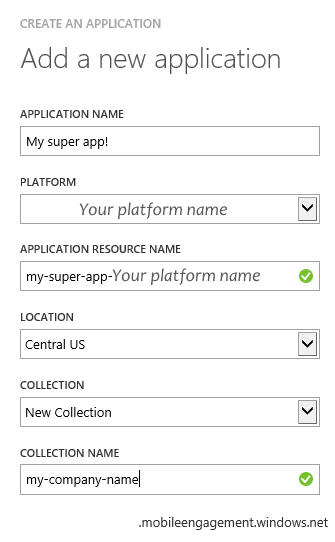
И к Job, и к Event можно добавить кастомные данные.

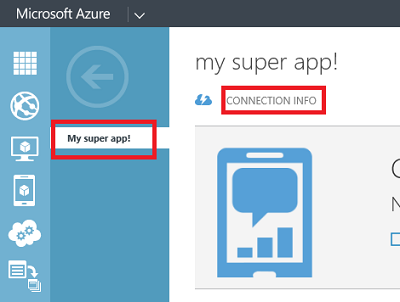
**Разработчику**

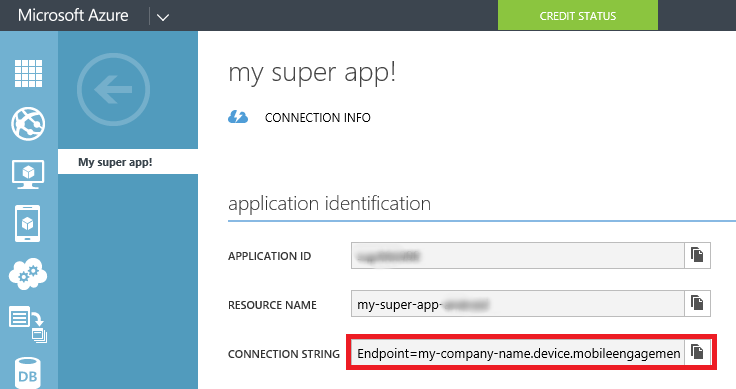
Предлагаю начать с того, что нужно знать разработчику, т.к. это быстрее и проще.

Создать новый сервис на портале (manage.windowsazure.com)

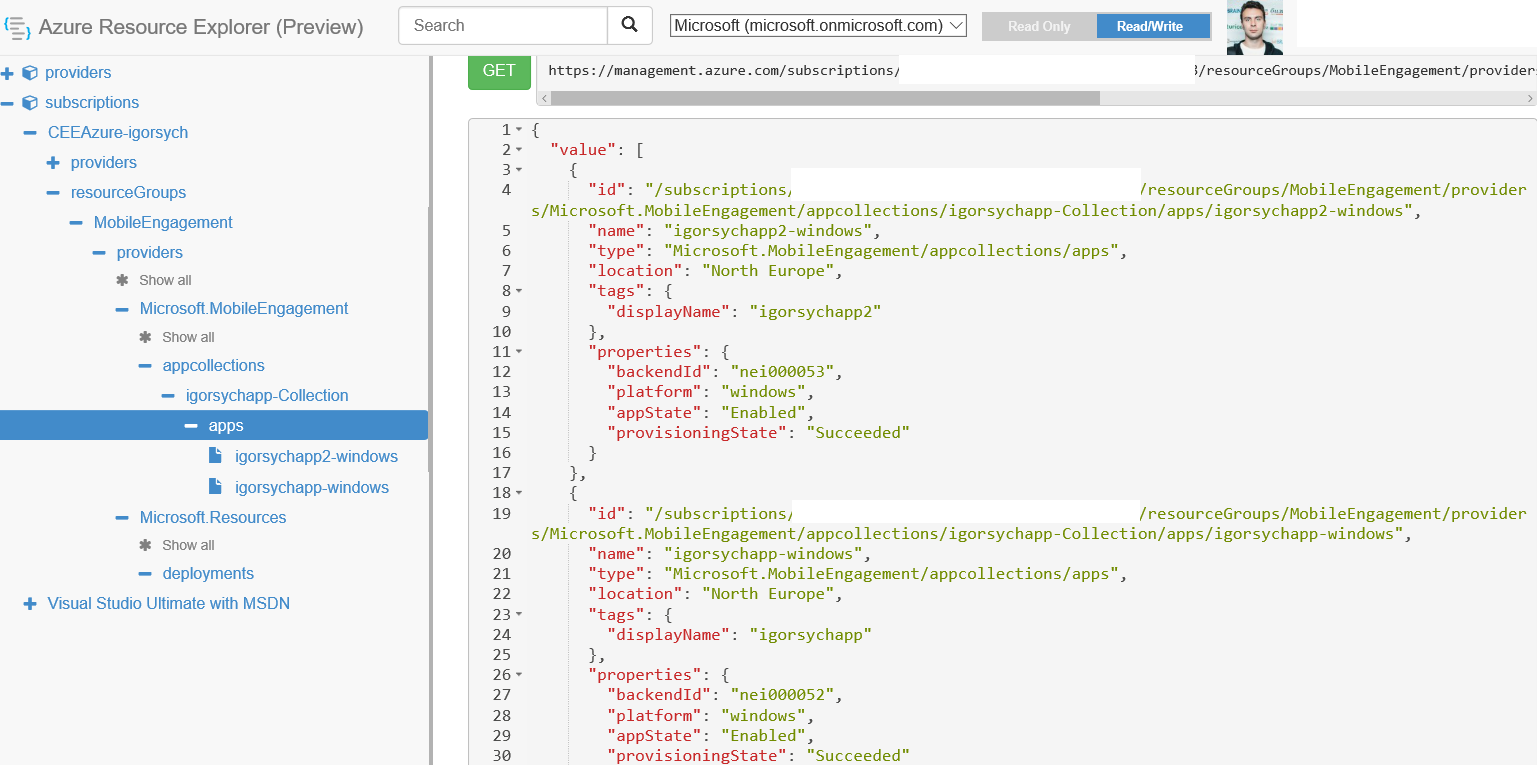








Хоть приложение и создается на classic портале, оно поддерживает ARM модель управления

****

А далее все зависит операционной системы/языка, под которую мы будем писать, в котором мы будем использовать ME. Готовые примеры интеграции можно взять с github <https://github.com/Azure/azure-mobile-engagement-samples>

Как и многие сервисы Azure, он не просто поддерживает IOS и Android, но и эти sdk были сделаны едва ли не раньше, чем под windows.

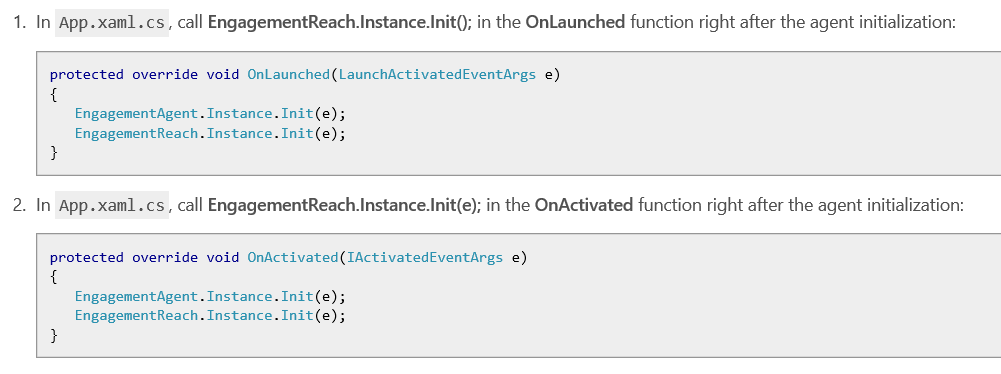
* ObjectiveC<https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-ios-get-started/>
* Swift <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-ios-swift-get-started/>
* Java <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-android-get-started/>

Для Windows платформы есть 2 мануала: для Silverlight приложений и для universal apps (8.1).

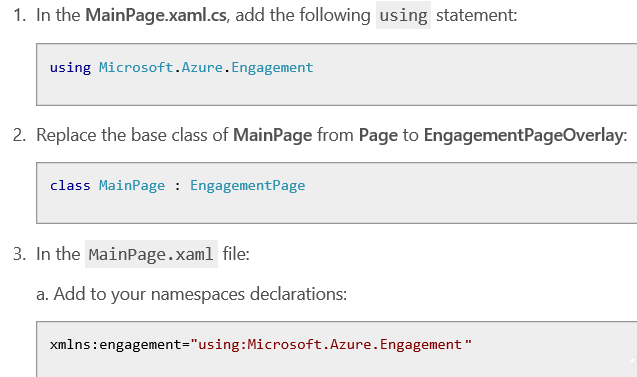
* <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-windows-store-dotnet-get-started/>
* <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-windows-phone-get-started/>
* На данный момент Windows 10 Universal инструкции нет, но это поддерживается. Просто берем документацию от 8.1 <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-windows-store-dotnet-get-started/>
* Для Xamarin уже начали писать sdk <https://feedback.azure.com/forums/285737-mobile-engagement/suggestions/10029978-xamarin-c-sdk-to-work-with-azure-mobile-engagem>

Сама SDK делится на 2 части: стандартная (сбор данных) и расширенная (отправка нотификаций). Приведенный ниже код для расширенной версии (т.к. базовая в нем тоже подключена).

В App.xaml.cs добавим инициализацию.



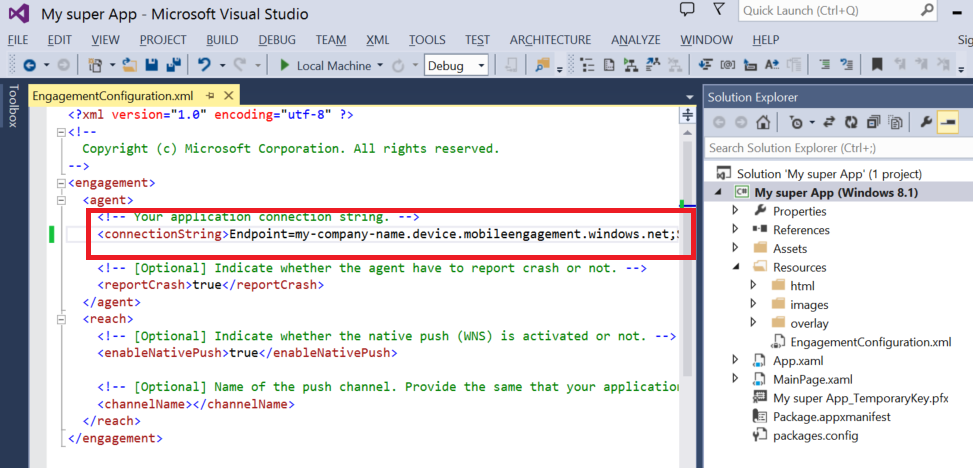
На каждой странице, где мы хотим включить в нашу аналитику, нужно заменить базовый класс.



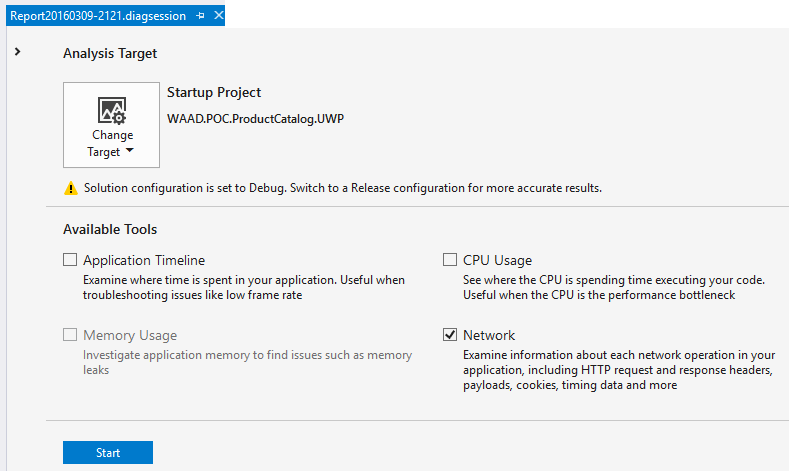
Если Вас такой вариант наследования не устраивает, то можно самим имплементировать функционал, который в базовых классах реализован. <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-windows-store-use-engagement-api/>

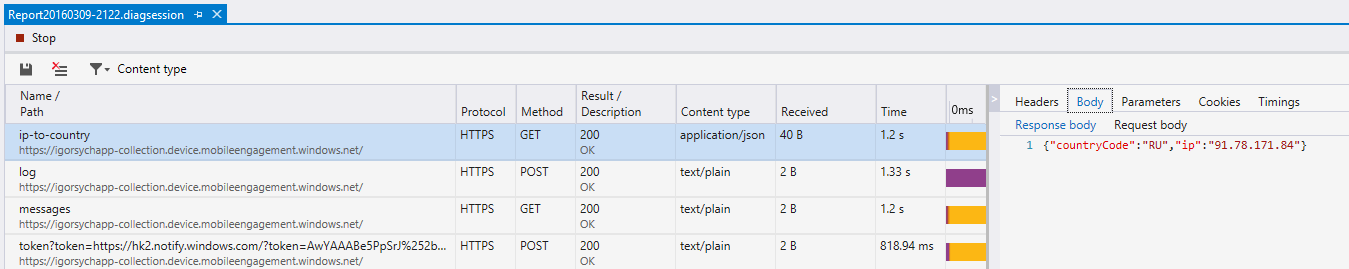


Ну, и нужно добавить конфигурацию. Для этого нужно добавить xml файл.



На этом базовые шаги по настройке окончены. Чтобы убедиться, что данные отправляются, предлагаю запустить Visual Studio с network profile и посмотреть, куда идут запросы.

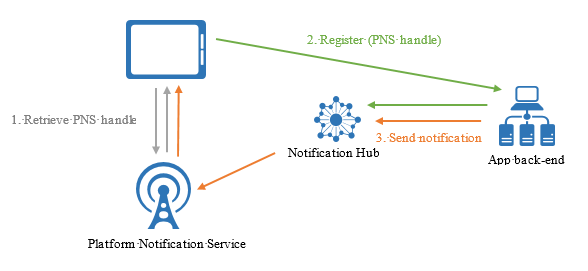




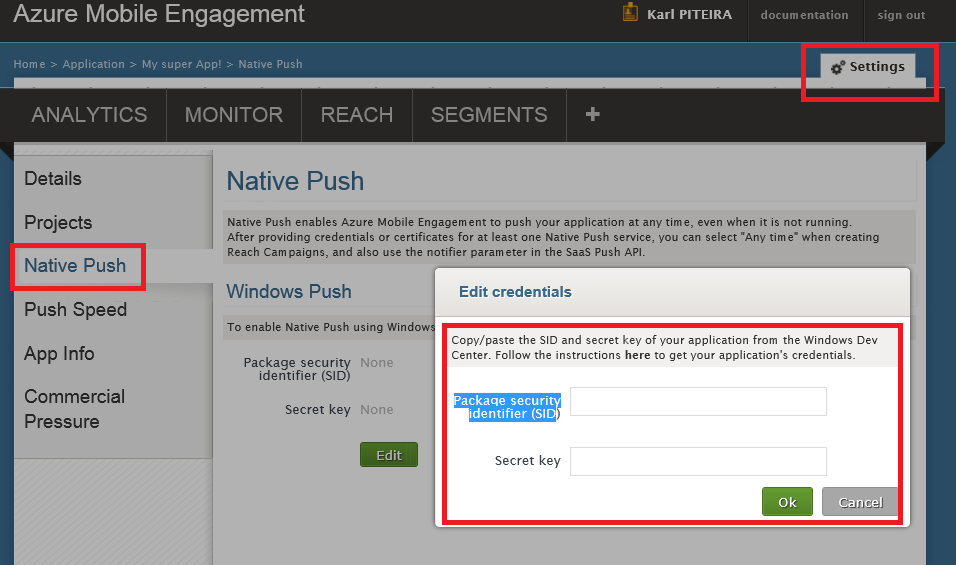
Важная настройка - как часто отправлять эти логи на сервер? По умолчанию, это происходит в real time режиме, но можно установить пакетную отправку. <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-windows-store-integrate-engagement/>



Перед тем, как переходить к аналитике, остается еще 1 важный вопрос: а как, собственно, нотификации-то доставлять, не голубиной же почтой? Для того, чтобы отправлять нотификации, используется Azure Notification Hub, но это для нас скрыто за Mobile Engagement.

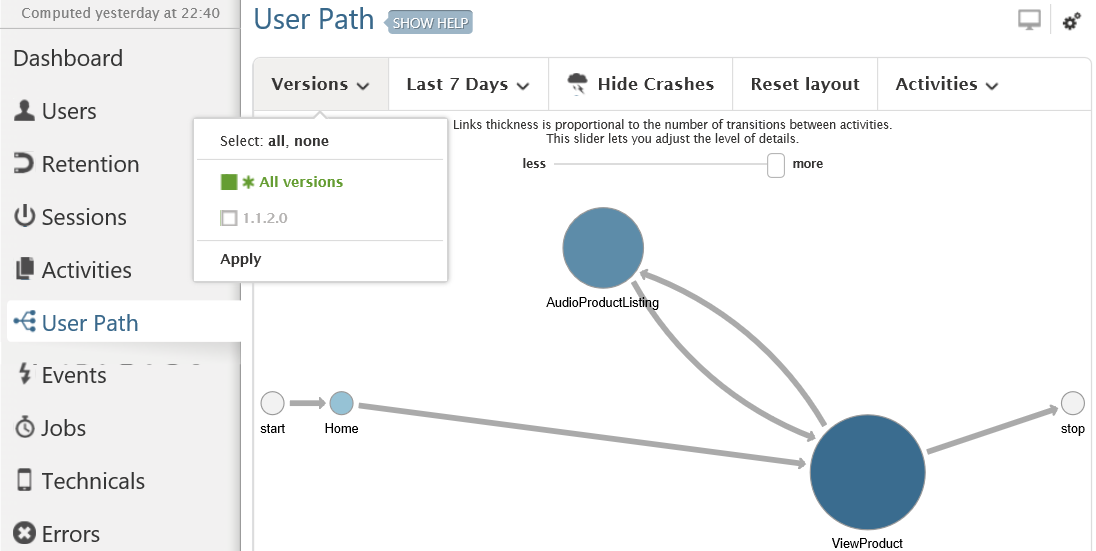


Как и для любых нотификаций, нам нужно будет зарезервировать имя приложения в windows store (или любом другом магазине), скопировать оттуда ApplicationID и userSecret и вставить их в Mobile Engagement.



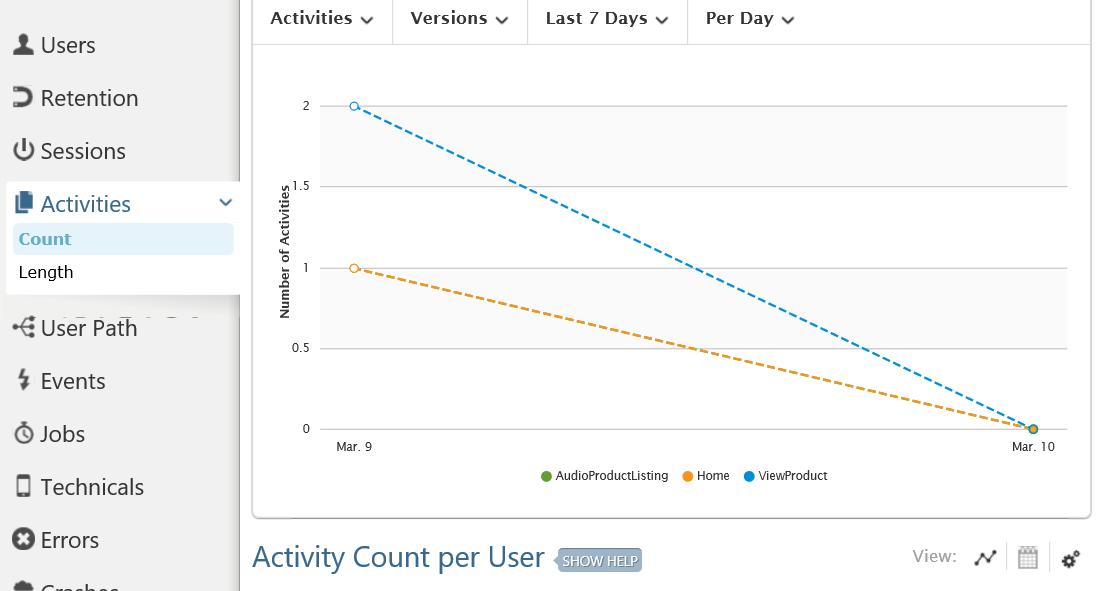
**Аналитика**

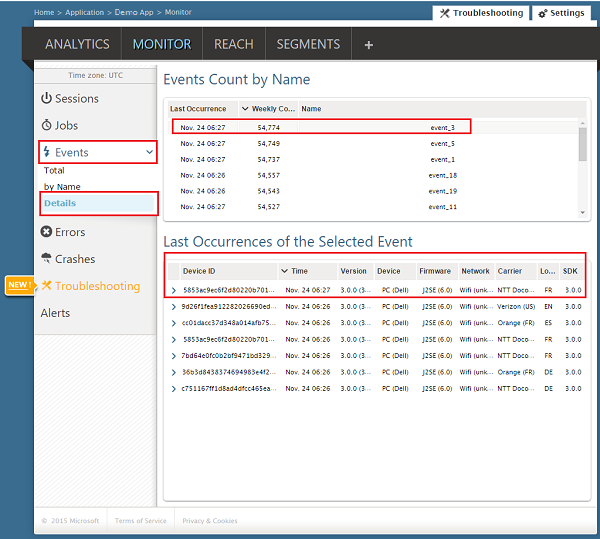
Прежде чем делать “Welcome Campaign” (“Hello World” в мире мобильной аналитики), хорошо бы, как минимум, посмотреть на аналитику, собираемую для вас. Статистика по количеству пользователей и сессий, уверен, понятна всем. Давайте начнем с UserPath



Это граф переходов между вашими страницами. Какие выводы можно из него сделать, которые нельзя сделать, посмотрев код и потыкав в приложение самому? Тут собраны те переходы, которые пользователи действительно использовали, а не все возможные. Статистика агрегирована по всем пользователям, и можно указать дополнительный фильтр для разных версий приложения.

Куда более интересна статистика по количеству активностей (открытия страниц).



Также интересно посмотреть последние события 

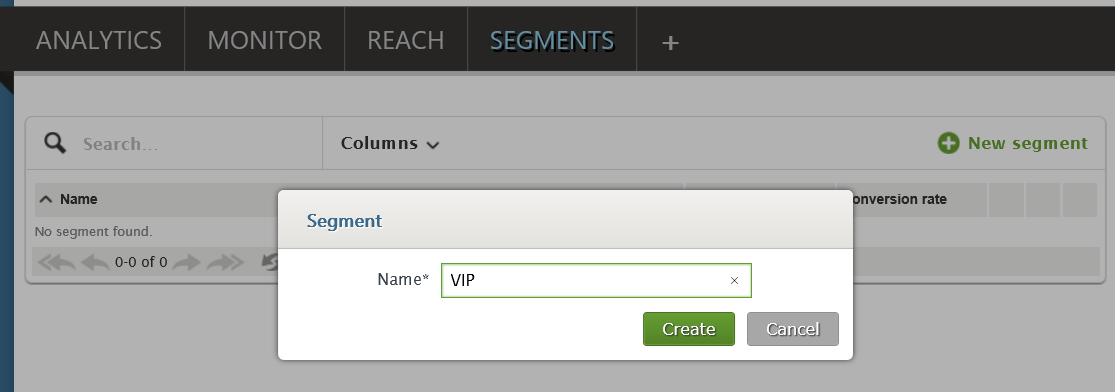
С собираемыми данными более-менее разобрались, давайте теперь перейдем к сегментации и нотификации пользователей.

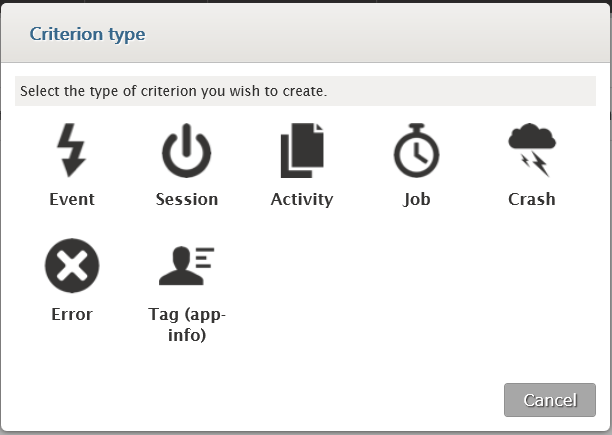
**Engage/Reach**

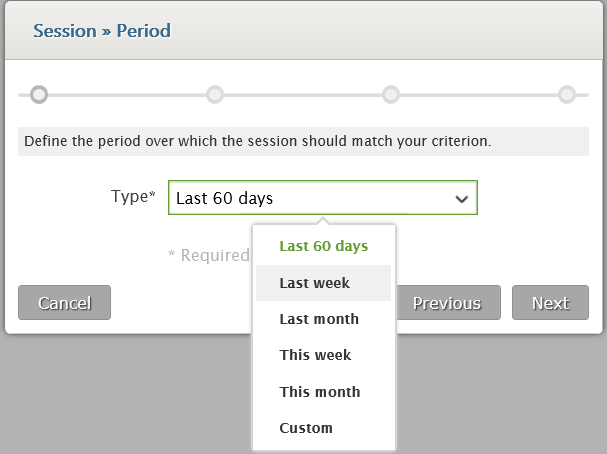
Прежде чем начинать спамить пользователей, было бы хорошо подумать над стратегией, над вашими kpi и т.п., чтобы не вызвать у пользователей лютую ненависть. <https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-define-your-mobile-engagement-strategy/>

Затем уже можно идти на портал и выбрать из всех пользователей вашего приложения сегменты пользователей по одному вам известному способу.

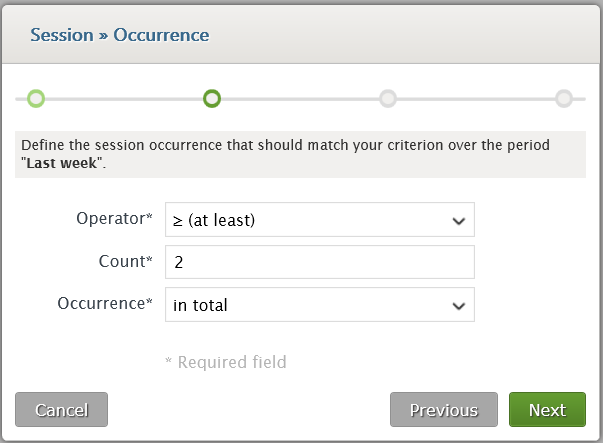
[**https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-user-interface-segments/**](https://azure.microsoft.com/en-us/documentation/articles/mobile-engagement-user-interface-segments/)

**Создаем новый сегмент **

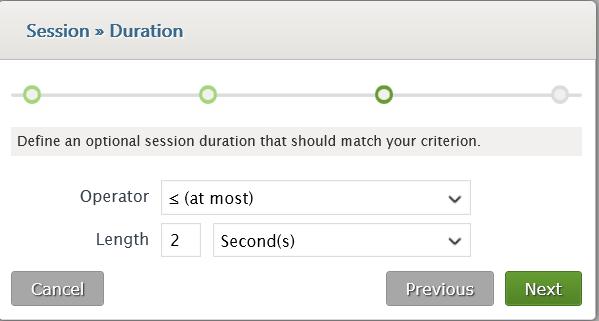
**И затем, шаг за шагом, выбираем критерии отбора пользователей. Этих критериев может быть много, но начнем с самого очевидного и общего - с пользовательских сессий.**

****

**Выбираем сессии за последние несколько дней.**

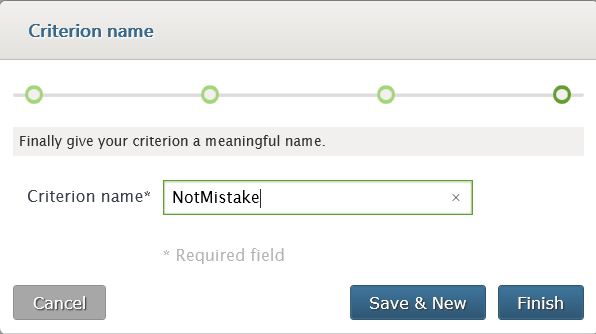
****

**У которых было хотя бы 2 разных операции (не просто открыл-закрыл случайно…а что-то сделал.)**

****

**И длительность сессии более n секунд (опять же фильтруем случайные открытия).**

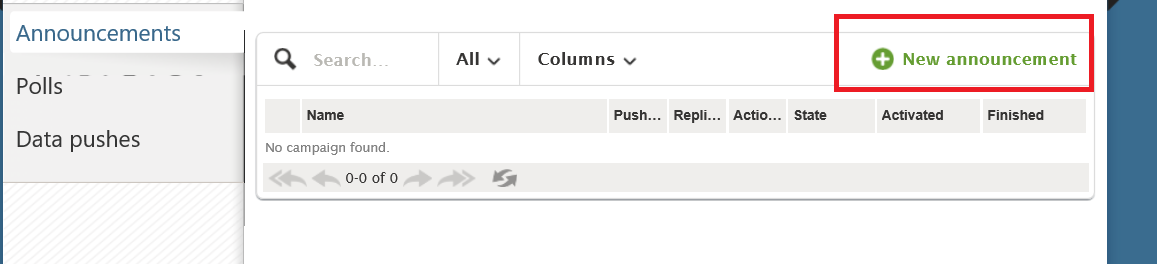
**Сохраняем**

****

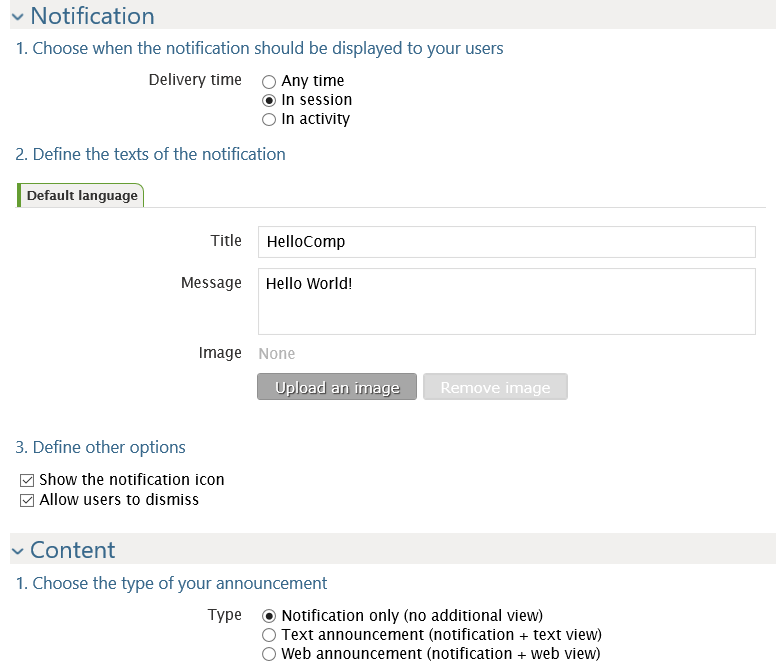
Мы создали наш первый сегмент. К сожалению, не все процессы в Mobile Engagement проходят в real time и нам придется подождать, прежде чем мы узнаем, сколько пользователей попадают под наши условия. А ждать придется до 24 часов. Согласен, что не самый интерактивный способ поэтому предлагаю проголосовать за эту идею на user voice <https://feedback.azure.com/forums/285737-mobile-engagement/suggestions/12868080-caculate-aggregates-statistics-every-6h-or-less>

**После того, как мы посмотрели на сегменты, можно уже и рассылкой заниматься.**

**Reach/Engage**

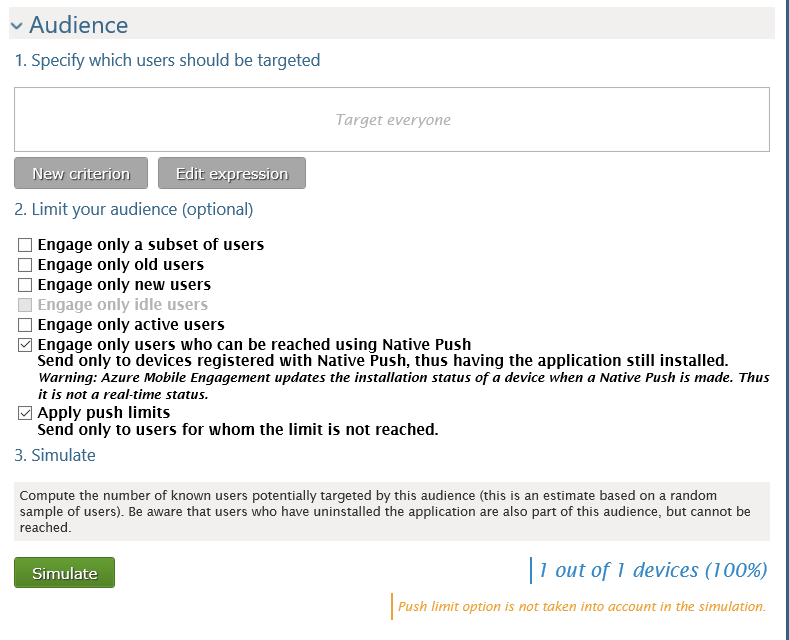
Давайте теперь создавать маркетинговую компанию на основе нотификаций:

Создаем компанию, указываем текст нотификации и выбираем кому будем отправлять:

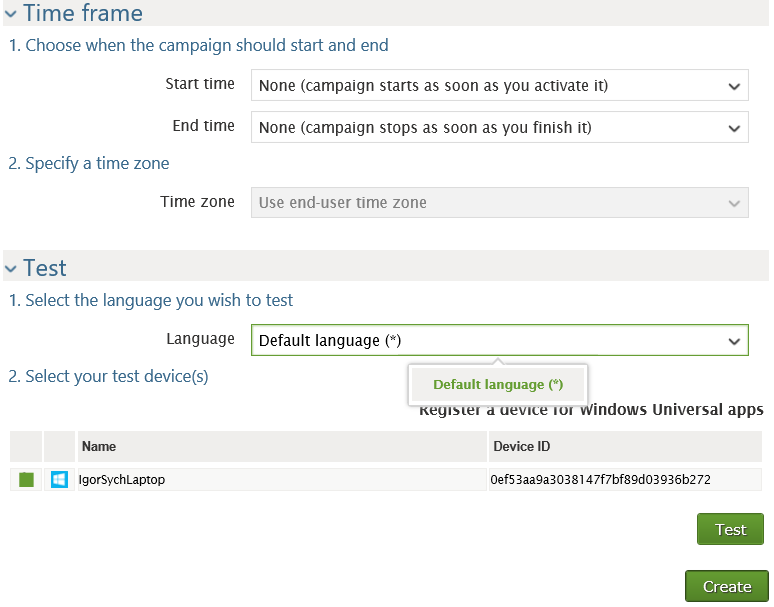


Тут мы выбрали, что мы только сделаем нотификацию с текстом Hello World.

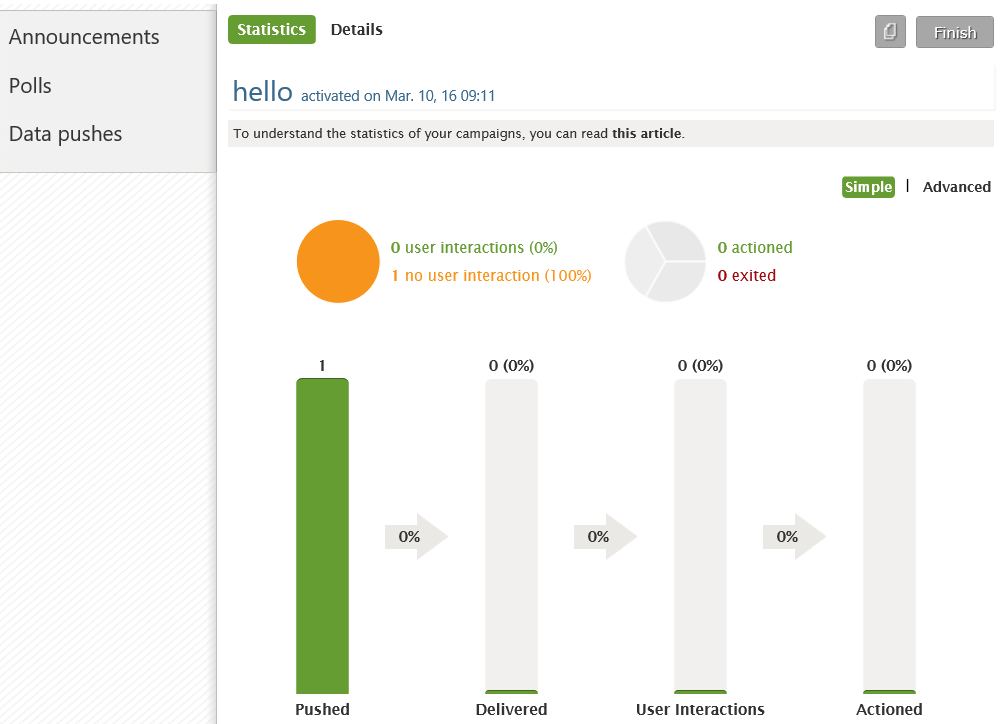
Затем мы выбираем кому мы отправляем. Если же мы нажмем кнопку simulate, мы узнаем сколько юзеров попало под эту выборку



Чтобы не спамить всех бесконечно, можно указать время действия компании.

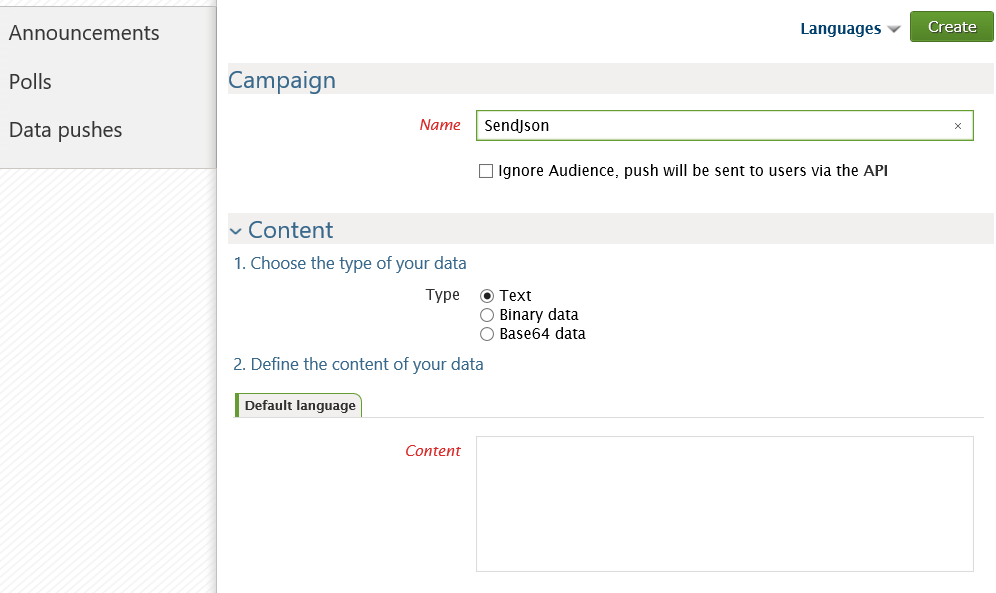


Как только компания создана и запущена, можно посмотреть, сколько нотификаций было доставлено, сколько просмотрено и т.п.



Какие еще способы взаимодействия с пользователем у Вас есть?

* Вы можете показать ему опросник…
* Вы можете со стороны сервера отправить на клиент данные (push data)



**Цены**

Как и любой сервис, mobile engagement имеет цену <https://azure.microsoft.com/en-us/pricing/details/mobile-engagement/>



Оплата по активным пользователям в месяц. Активный пользователь - пользователь открывавший ваше приложение хотя бы раз за месяц.

**Summary:**

р