

Запуск приложений платформы

Модуль запуска тренажера в MEP:

- Unity 3D Launch SDK - <https://cloud.modumlab.com/s/kZJ93FkfzJ87YL7>

Основные положения

Запуск осуществляется посредством перехода по url, схема которого зарегистрирована в ОС и ассоциирована с приложением.

Стандарт для bundle id

Приложения

Для всех приложений платформы (т.е. кроме исполняемых на заказ специфично под заказчика) bundle id должен иметь вид:

- **com.<companyname>.mer.app.<app_name>**

где:

- <companyname> - наименование компании, требования аналогично <app_name> (см. ниже);
- <app_name> - краткое наименование приложения, удовлетворяющее следующим требованиям:

1. Только нижний регистр:
 - a. правильно: video360player
 - b. неправильно: Video360PLayer
2. Состоит из букв английского алфавита:
 - a. правильно: video360player
 - b. неправильно: видео360проигрыватель
3. Всегда начинается с буквы:
 - a. правильно: video360player
 - b. неправильно: 360videoplayer
4. Не содержит любых других символов типа подчеркивания или дефиса:
 - a. правильно: video360player
 - b. неправильно: video_360-player

В случае сложного составного приложения следует писать название без разделителей, например "networkscenariosclient".

Правила обусловлены необходимостью унификации между платформами, например, iOS позволяет использовать дефис, а Android - нет, Android позволяет писать подчеркивание, а iOS нет. Соответственно, выбирается вариант, подходящий для всех платформ.

Данные от лаунчера платформы передаются в приложение в виде URI

Scheme - описывает конкретное приложение (с каждым приложением должна ассоциироваться уникальная url схема)

Host - описывает действие, которое необходимо произвести (сейчас всегда "launch")

Path - описывает тип продукта (одно приложение может воспроизводить разные типы продуктов)

Параметры запуска (URL query)

Для передачи данных приложению используются параметры url, например:

- `com.modumlab.mer.app.myapp://launch/simulation/?param1=value¶m2=value2`

Вводятся следующие стандартные параметры:

1. **product** - данные о запуске продукта, включая список контента;
2. **analytics** - данные, необходимые для доступа к публичному api аналитики.

Значения параметров упаковываются в json, ниже приведен формат данных стандартных параметров.

Product

Данные о запуске продукта.

Формат:

```
product={
  "identity": {
    "type": "video360",
    "id": "prod_video360_3fbc7ac44b784654b2d494bbfa20f598"
  },
  "about": {
    "title": "Видео 360",
    "description": "Видео 360."
```

```

},
"app_version": {
  "min_version": {
    "code": 1,
    "display_name": "1.0"
  }
},
"content": {
  "list": [
    {
      "content_id": "cont_video360_ed5672bda034ede62ac7f4bb3055ef0b",
      "content_type": "video/video360",
      "mime_type": "video/mp4",
      "hash": {
        "sha256": "56j75eh456g456gw456g"
      },
      "size": 12998775,
      "filename": "smesharikis04e21_fin3_final-cut_10v2-oleg.mp4",
      "metadata": {
        "video_id": "видео_про_смешариков_s04e21_fin(3)"
      }
    },
    ...
  ]
},
"metadata": {
  "creator_account_id": "user_default__9f9a1bc3265d41be89aef5202f5b43f3",
  "time_frame": false,
  "launch_args": {
    "scene_name": "q2dm1",
    "mode": "deathmatch"
  },
}
}

```

Пояснения по списку контента:

1. **content.list** - список контента продукта:
 - a. **content_type** - МЕР-тип контента;
 - b. **path** - полный путь и имя файла (расширение может не быть указано);
 - i. **⚠️ Внимание!** После скачки лаунчером, контент располагается в некой папке на устройстве под именем, соответствующем sha256-хэшу файла и с оригинальным расширением. Соответственно, идентифицировать файл по имени не представляется возможным. Следует использовать либо поле "filename", либо отдельную запись в метаданных (поле "metadata") для идентификации нужного файла среди прочих в списке контента. Предпочтительно второе (использование ключа в метаданных, см. пример выше).
 - c. **filename** - имя файла с расширением;
 - d. **metadata** - метаданные контента - json-объект с произвольными данными, вносимыми при формировании списка контента на серверной части:

- i. Здесь можно указывать дополнительные идентификаторы и/или метаданные, относящиеся к данному файлу;
- ii. Формат и содержимое метаданных определяется отдельно для каждого приложения (подлежит согласованию).

Analytics

Данные, необходимые для доступа к публичному API аналитики

Формат:

```
{  
  "endpoint": "https://analytics.example.com",  
  "analytics_token": "w3495h4qp934gdow364fv23423423423yhh167io9ohvvq2o"  
}
```

Процесс запуска приложения

В Windows существует ограничение на длину передаваемых аргументов. Для передачи полного uri в приложение используется named pipe.

Имя named pipe в приложение передается через аргумент командной строки следующим образом:

```
"runnable.exe --  
url=com.modumlab.mep.app.myapplication://launch/simulation/?pipe=LaunchPipe_1231456",
```

где LaunchPipe_1231456 это имя named pipe.

Приложение в процессе запуска должно считать из pipe полный uri и обработать его согласно описанию выше.

Таймаут операции - 90 секунд.

Вид полного uri - следующий

```
"com.modumlab.mep.app.myapplication://launch/simulation/?analytics={"endpoint"  
: "https://analytics.example.com",  
"analytics_token": "w3495h4qp934gdow364fv23423423423yhh167io9ohvvq2o"}"
```

Архив и манифест

Windows приложение, загружаемое для работы на платформе должно быть оформлено в виде zip архива, содержащего все необходимые для работы файлы.

В корень архива необходимо добавить файл с названием "StandaloneManifest.json", содержащий ряд метаданных.

```
{  
  "meta": {  
    "manifest_version": 1  
  },  
  "application": {  
    "display_name": "My Application Name",  
    "bundle_id": "com.modumlab.mep.app.myapplication",  
    "executable": "runnable.exe"  
  },  
  "version": {
```

```
"display_name": "1.0",
"code": 1
},
"launch": {
  "url_scheme": {
    "host": "com.modumlab.mep.app.myapplication"
  }
}
}
```

Параметр `executable` указывает относительный путь до исполняемого файла приложения.

Version -> code указывает номерную версию приложения, поле должно увеличиваться при каждом обновлении приложения.

Дополнительные параметры запуска

Запуск при проигрывании

В корне архива может быть расположен файл с названием "LaunchArgs.json". В таком случае его содержимое (при условии, что это корректный json) будет передано в рамках структуры параметра запуска "product" (см. раздел "Параметры запуска > Product"):
product.metadata.launch_args.

Пример "LaunchArgs.json":

```
{
  "scene_name": "q2dm1",
  "mode": "deathmatch"
}
```