

Лабораторная работа № 2. Проект – разработка простого AR-Приложения для Android-устройства (смартфон, планшет и пр.). Создание в графическом редакторе Unity 3D сцены дополненной реальности: **визуализация 3D - модели.**

Объекты ДР – это объекты проекта, создаваемого с помощью средств платформы Vuforia.

Введение.

Работа по созданию приложений ДР заключается в заведении проекта и объектов проекта (Контент) в **Vuforia**, а разработка 3D-сцен для объектов этого проекта осуществляется в **Unity 3D**. При этом **Vuforia** отвечает за идентификацию проекта через **License key** (см. ниже), а привязка к будущей сцене виртуального 2D- или 3D-объекта (например, 3D-модели, плоских изображений, видеоклипов и пр.) будет осуществляться через определяемую в **Vuforia** метку (**Target**). Допустимые в используемой в Лабораторном практикуме версии **Vuforia Engine** типы таргетов были подробно рассмотрены в Описании ЛР№2, Часть 1.

ВАЖНО!! → вся работа с **Vuforia** (с проектом, объектами) осуществляется через **web-интерфейс**, иными словами, **Vuforia** является облачным приложением. А работа с **Unity-3D** осуществляется непосредственно на компьютере разработчика, т.е. локально.

Связь между облачным ведением проекта (в **Vuforia**) и локальной проработкой сцен Приложения ДР должна быть выполнена за счет импорта подготовленных объектов проекта из облака **Vuforia** в среду редактора **Unity-3D**.

Рассмотрим типовую процедуру создания простого (игрового) **AR Приложения**.

Предлагается разработать приложение ДР для Android-устройств, в котором при наведении камеры устройства на реальную метку (таргет – изображение, например, на бумаге, или на дисплее) пользователь в области воспроизведения на экране мобильного устройства (МУ) увидит другое, заранее подготовленную **3D – модель**.

Предварительные условия для начала работы:

- Интернет-соединение локального компьютера;
- Наличие аккаунта пользователя **Vuforia** (результат успешного выполнения ЛР №1);
- Установленная на компьютере разработчика система **Unity 3D** (результат успешного выполнения ЛР №1);
- Заранее подготовленные изображения для метки (таргета) и контент (**3D-модель**);
- Кроме того, для формирования собственно приложения ДР на машине разработчика (**MSWindows 10/8.1/7**) необходимо установить **Android SDK** – набор инструментальных средств для сборки **Android-Приложений** в чужой операционной системе. Для корректной установки **Android SDK** надо предварительно установить (**если это еще не сделано**) **Java Development Kit – JDK**, предпочтительно последней версии; адреса для скачивания:

Android SDK – <https://developer.android.com/about/versions/10/setup-sdk>

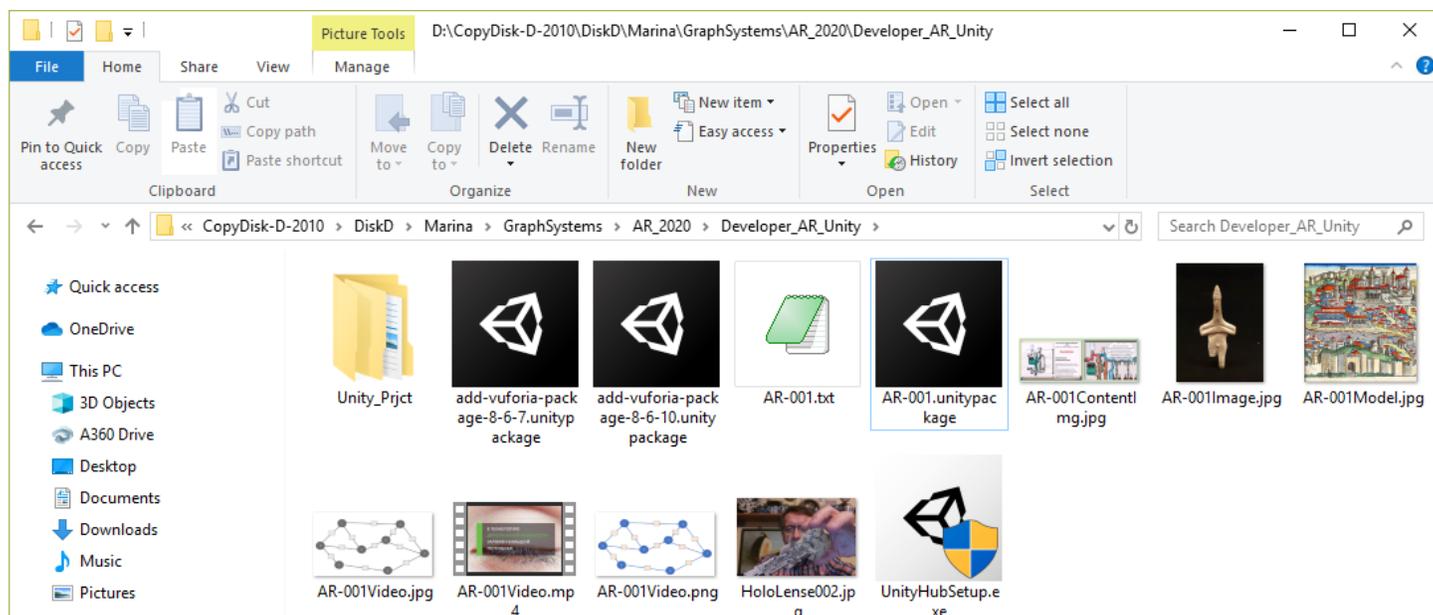
JDK - <https://www.java.com/ru/download/manual.jsp>

*Дополнительные пояснения по особенностям установки **Android SDK** под **Windows** – см. **Приложение 1**: http://a0601.narod.ru/ADD01_LR_02.pdf

Пункты 1, 2 выполнения ЛР: На сайте <https://developer.vuforia.com/> осуществляем вход с заведенными ранее логином/паролем для получения **лицензионного кода** и заведении **таргета** (генерация меток). Все шаги и возможные проблемы/особенности выполнения этих шагов – см. **Описание ЛР№2** (части 1 и 2).

В данном описании в результате выполнения пунктов 1 и 2:

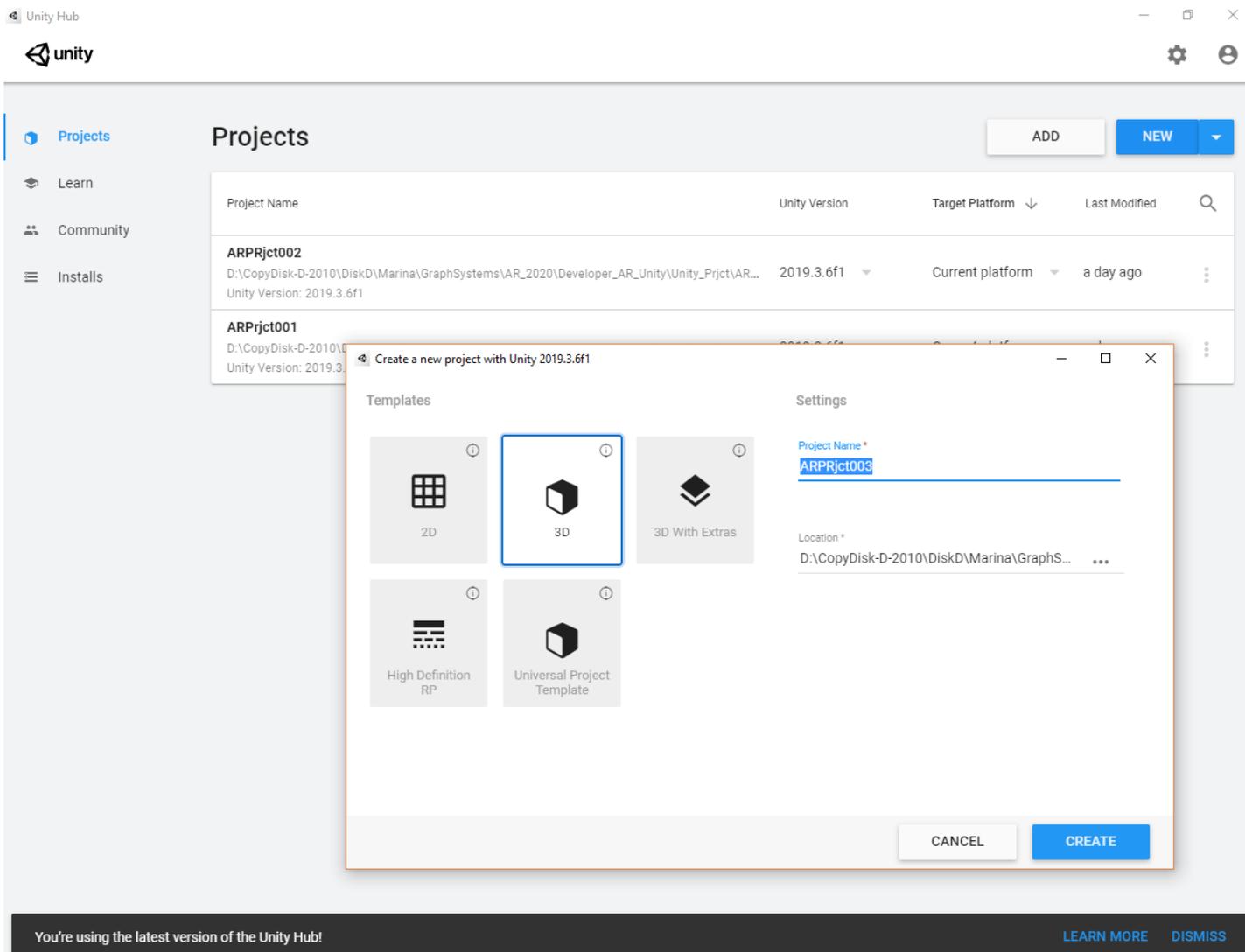
- Получен лицензионный код (можно использовать полученный ранее); продолжаем работать в том же проекте **Vuforia – AR001**
- В БД таргетов добавлен новый таргет **AR-001Model.jpg**.



Шаги выполнения ЛР №2, часть 3 подобны тем же шагам в ЛР №2, часть 1, 2, особенности и различия находятся в разделе 6.

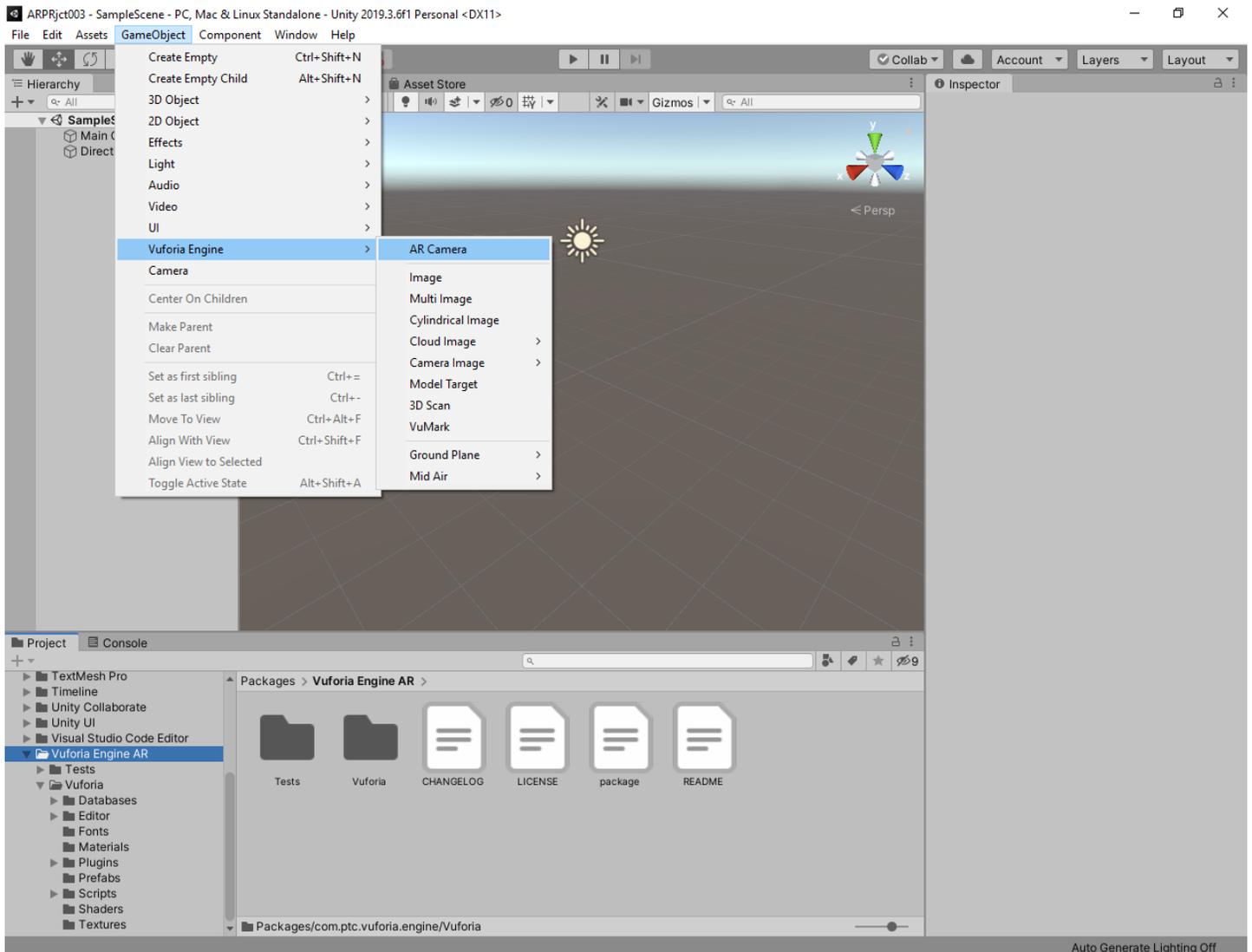
Пункт 3 выполнения ЛР. Стартуем приложение **Unity 3D**. (См. **Описание ЛР №2, часть 2**)

В данном **Описании ЛР №2, часть 3** предлагается создать новый проект с именем **ARPRjct003**:



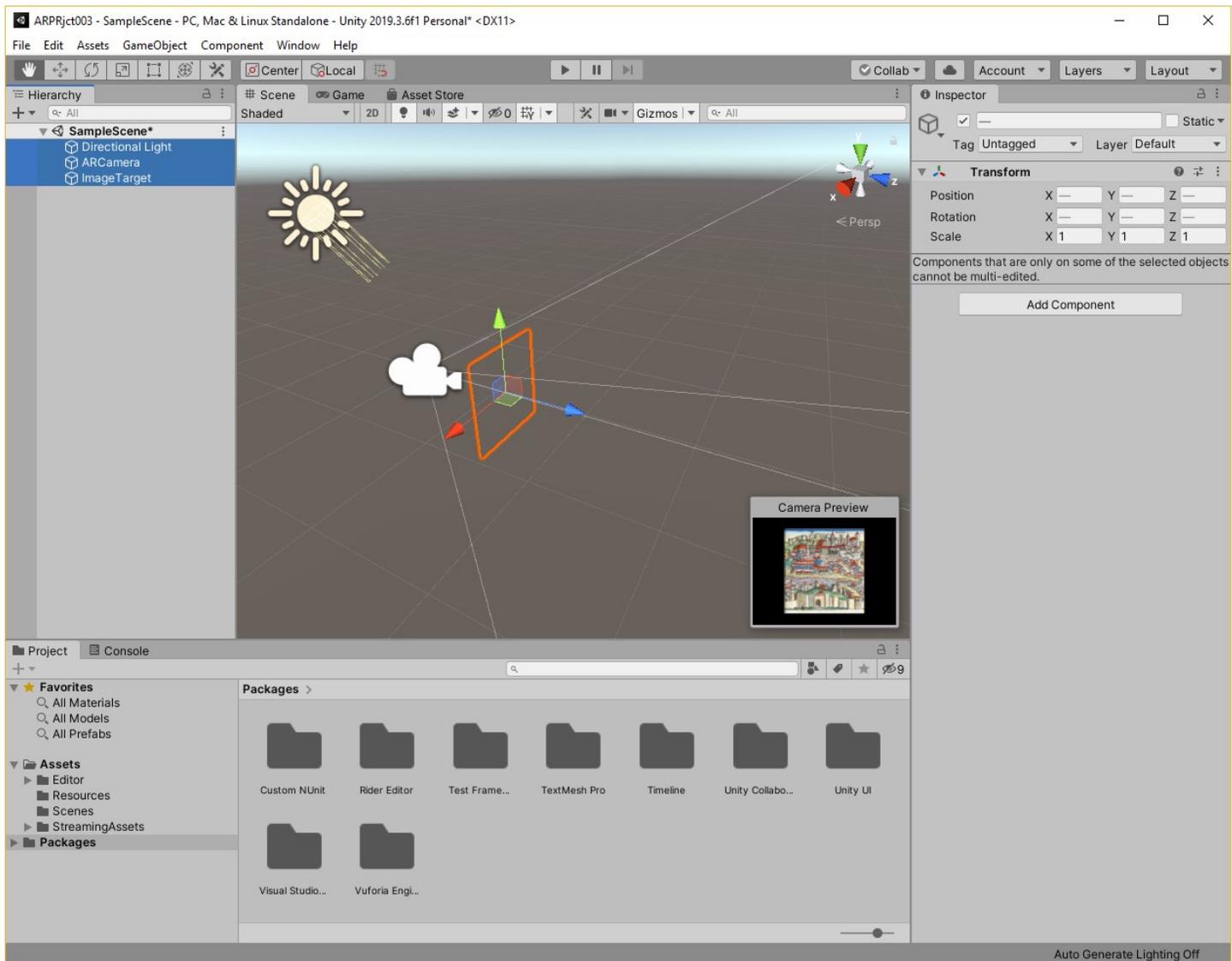
Дальнейшие шаги в новом Проекте в Пункте 3 – см. **Описание ЛР№2, часть 2.**

В результате выполнения этих шагов установлены Базовые настройки для работы **Unity 3D** с **Vuforia**:



Пункты 4, 5 выполнения ЛР. Создание Приложения ДР в среде **Vuforia + Unity 3D** для просмотра **3D-модели**. (См. **Описание ЛР №2, часть 2**)

- Через **GameObject Vuforia Engine**→устанавливаем **AR Camera**;
- Через **AR Camera** устанавливаем лицензию **Vuforia** (лицензионный ключ, полученный/сохраненный ранее);
- Удаляем **Main Camera**;
- В проекте, через **GameObject Vuforia Engine**→**Image** устанавливаем таргет;



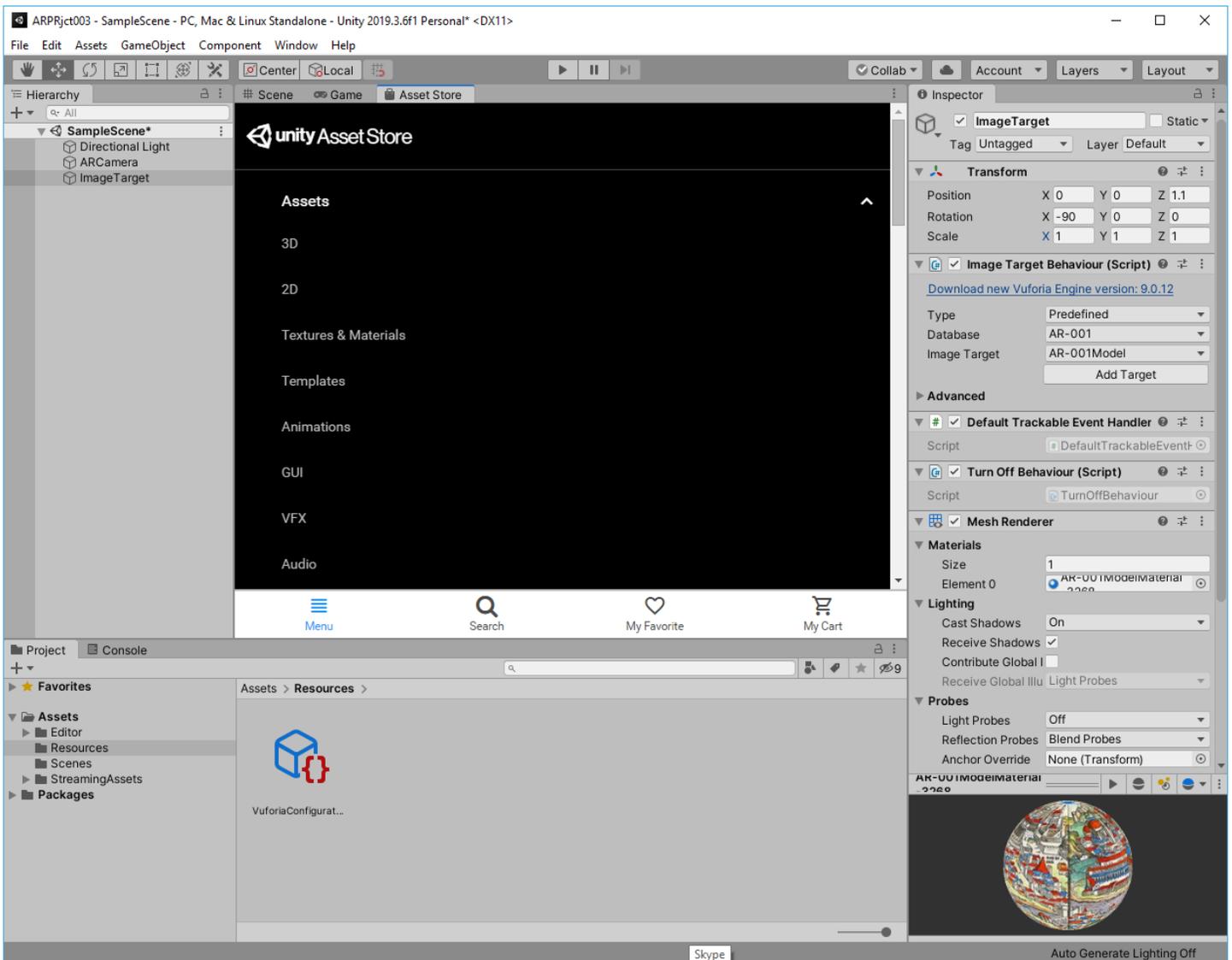
Теперь надо разместить контент, связанный с меткой, в сцене ДР.

Пункт 6 выполнения ЛР. Теперь необходимо загрузить 3D - Модель, которая должна появиться на экране устройства поверх изображения транслируемой реальности на месте, определяемом таргетом. Познакомьтесь с документацией:

<https://docs.unity3d.com/ru/current/Manual/HOWTO-importObject.html>

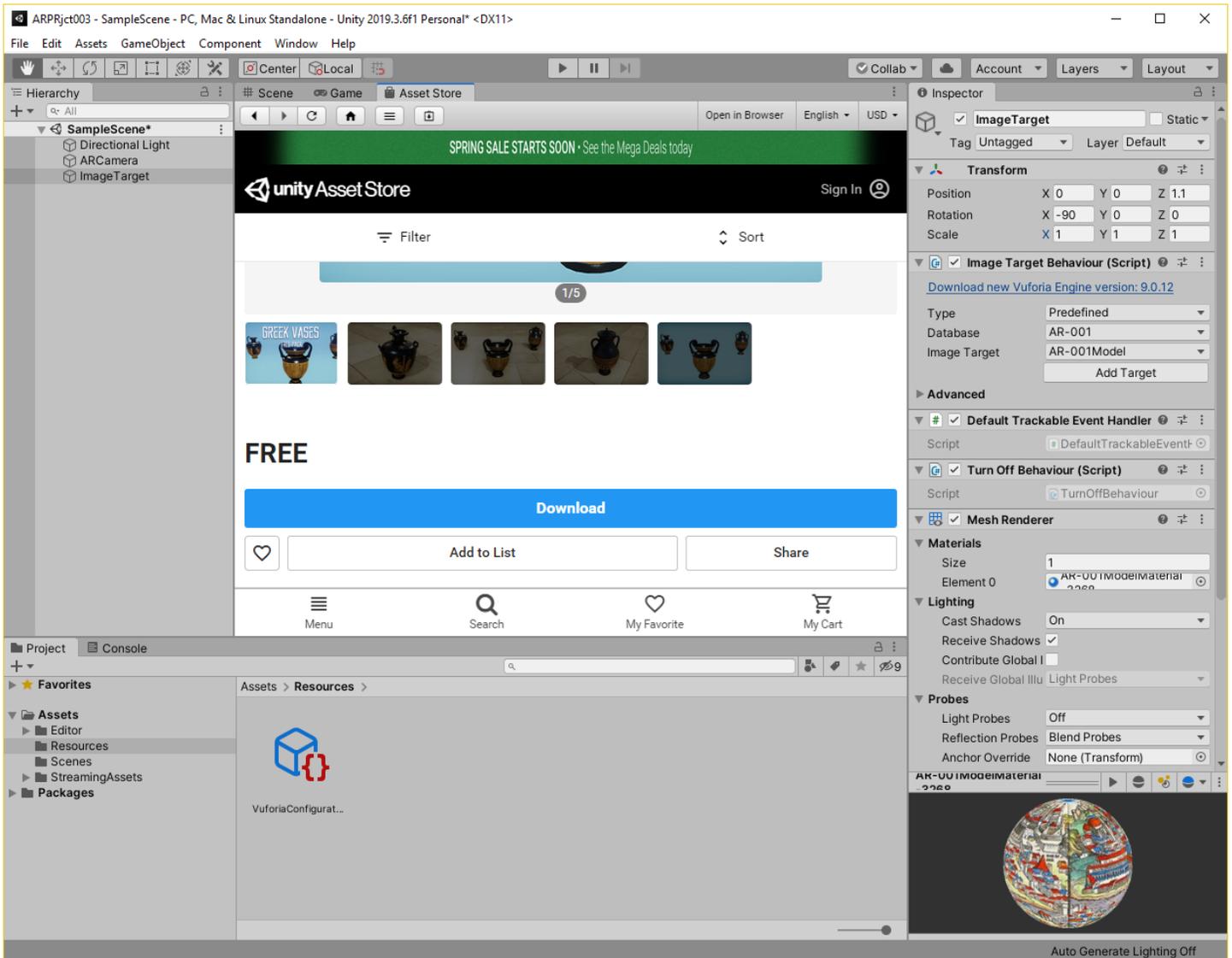
Сами **3D-Модели** могут быть импортированы в среду проектирования Приложения ДР из разных источников – внешних приложений (например, САПР); из библиотек готовых моделей, доступ к которым реализуется непосредственно из редактора **Unity 3D**. В нашем случае воспользуемся именно этим вариантом.

В 3-ей части ЛР №2 модели в среду **Unity Editor** будем добавлять из библиотек **3D-моделей Asset Store** («магазин»), доступных через интернет. Для этого переходим на одноименную вкладку **Asset Store** редактора **Unity 3D** и выбираем требуемый объект из открывшегося интерфейса «магазина».

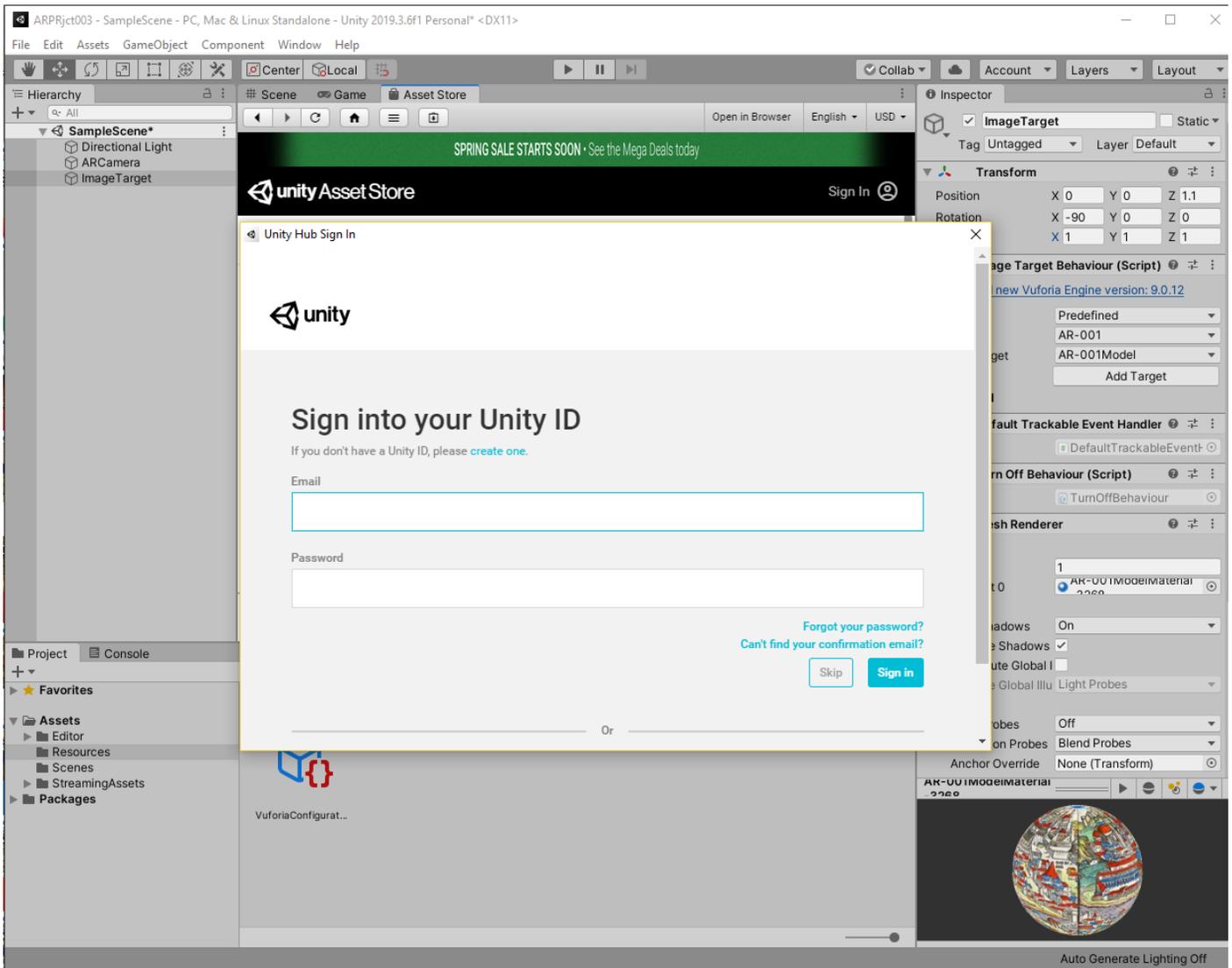


Предлагается выбирать бесплатные **3D-модели** из какой-либо области, воспользовавшись доступными (например, по стоимости) и формируемыми пользователем фильтрами. Например, бесплатную модель греческих ваз от разработчика с никнеймом **Nokobot**:

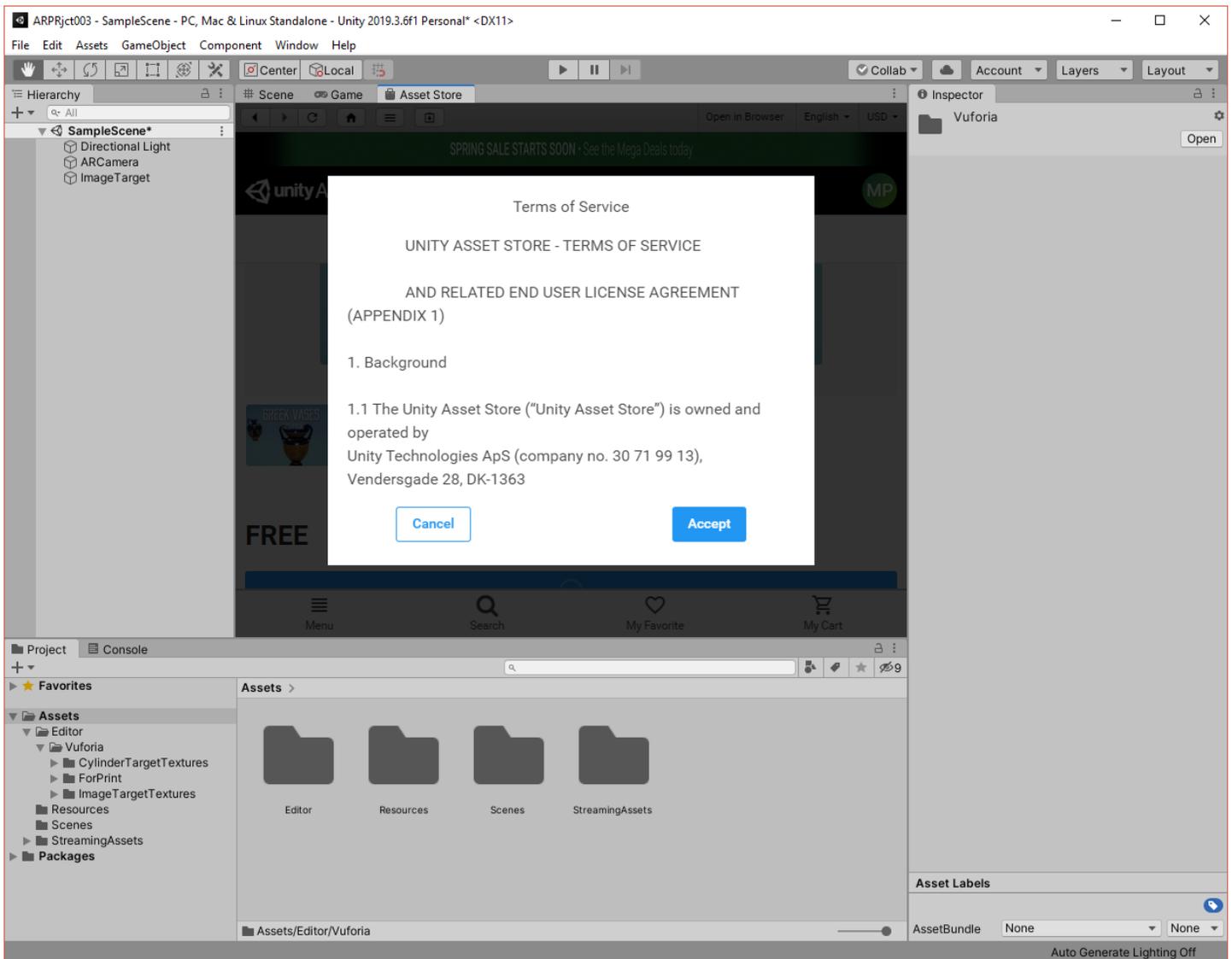




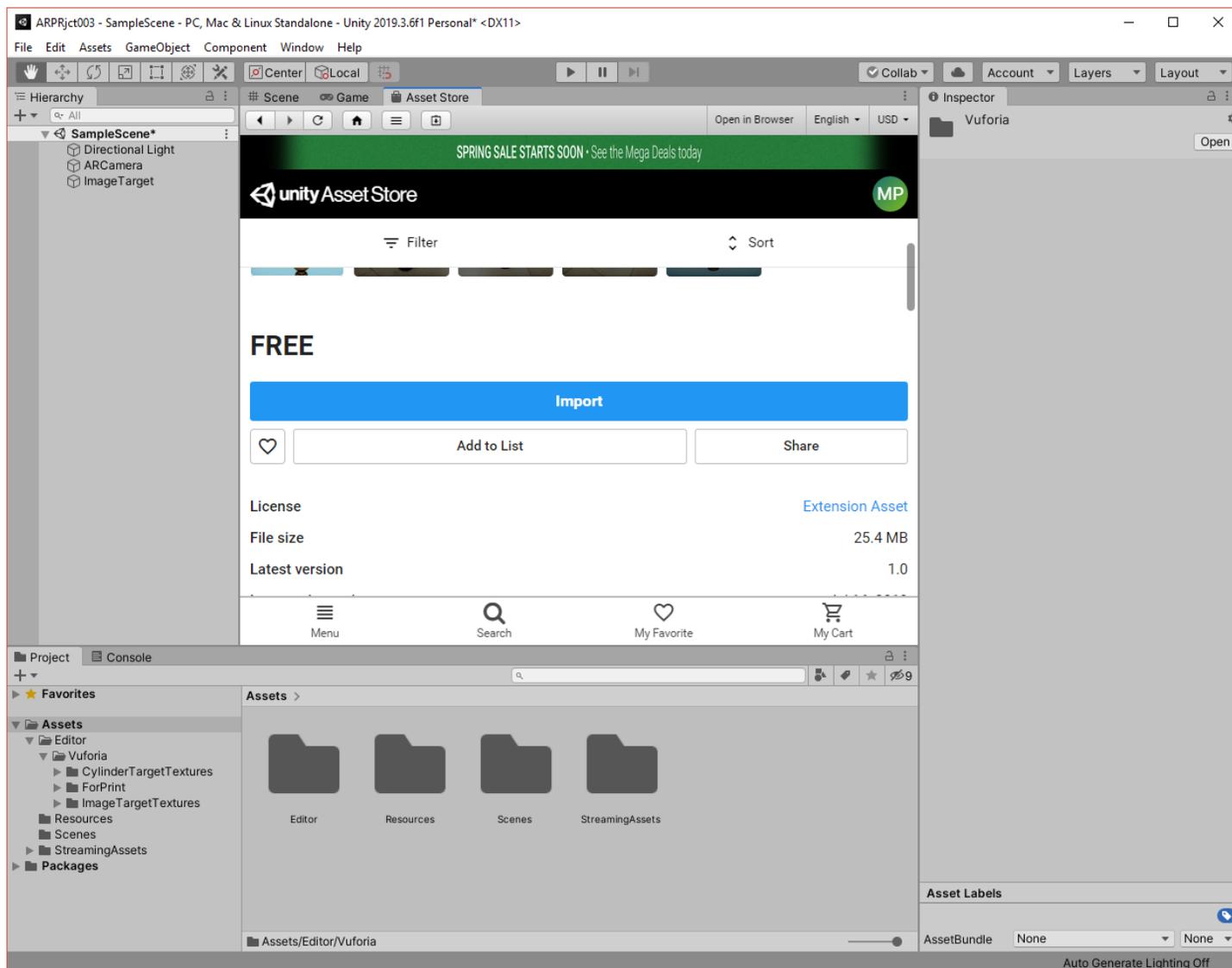
Нажимаем клавишу **Download**, после чего необходимо пройти процедуру аутентификации в **Unity 3D**:



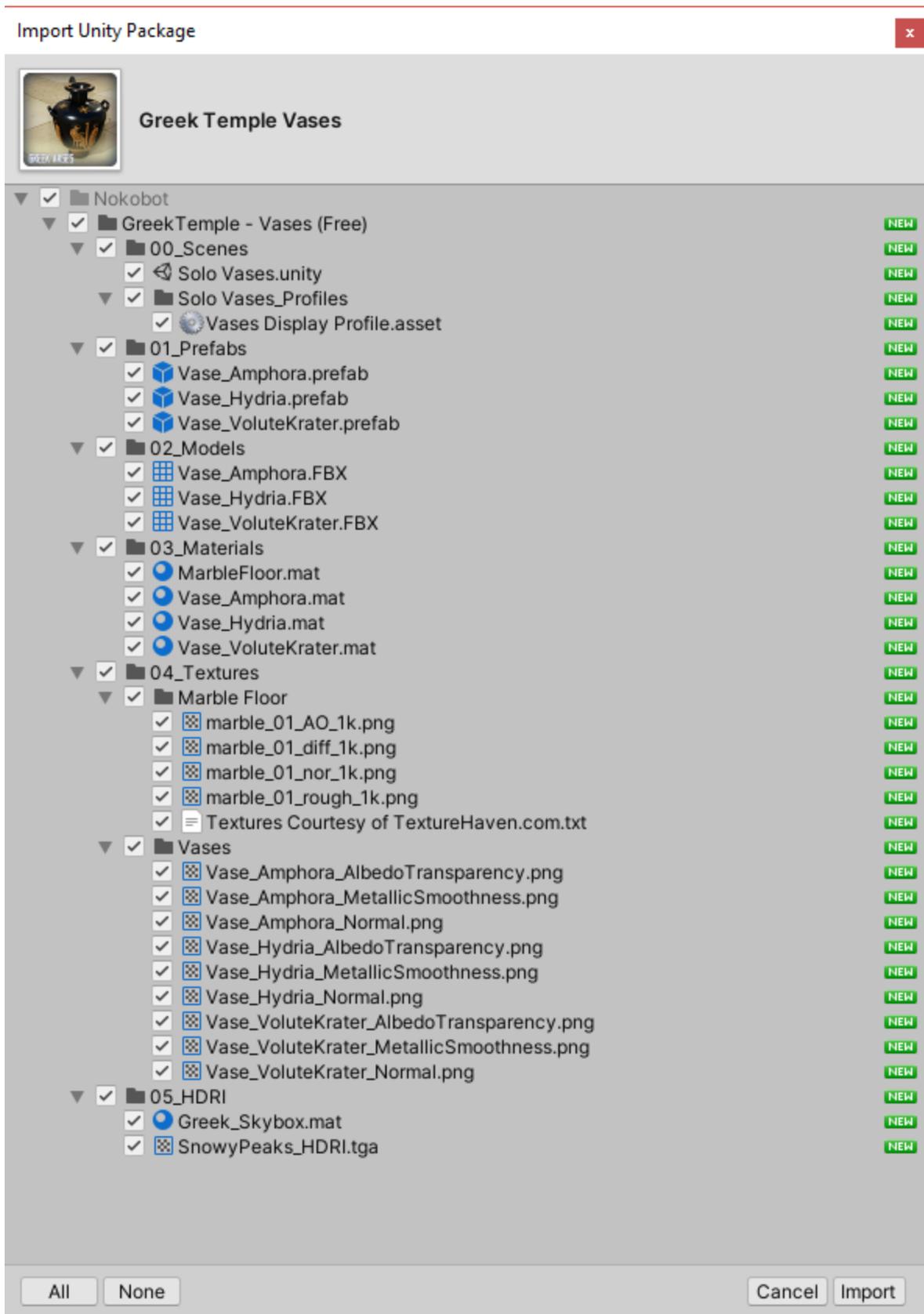
После успешного прохождения процедуры аутентификации еще раз нажимаем клавишу **Download**, соглашаемся с правилами использования выбранной модели в **Unity 3D**:



После чего начинается загрузка, по окончании которой выбранную модель надо импортировать:

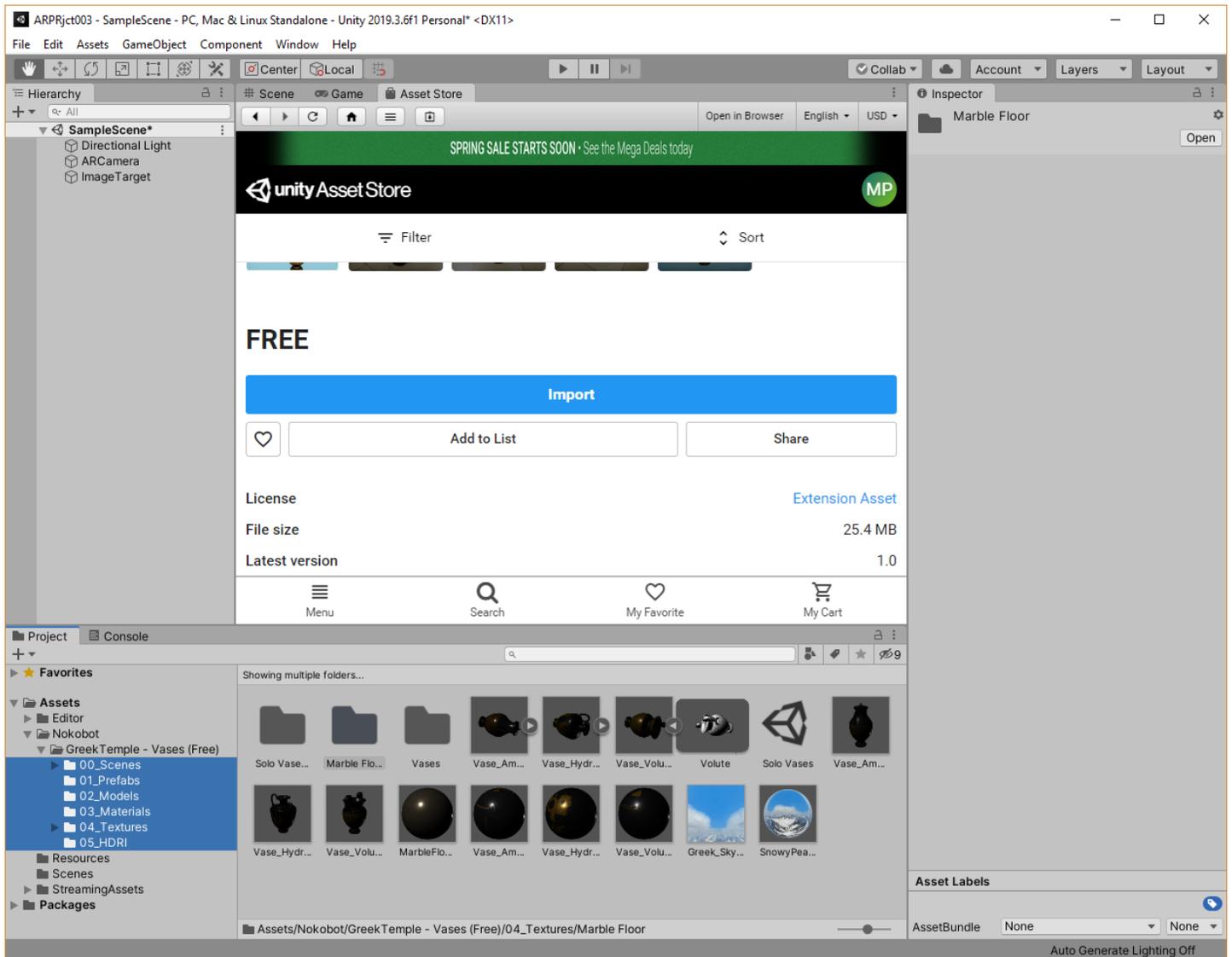


Подготовленная для импорта структура отображается в специальном окне **Import Unity Package**:

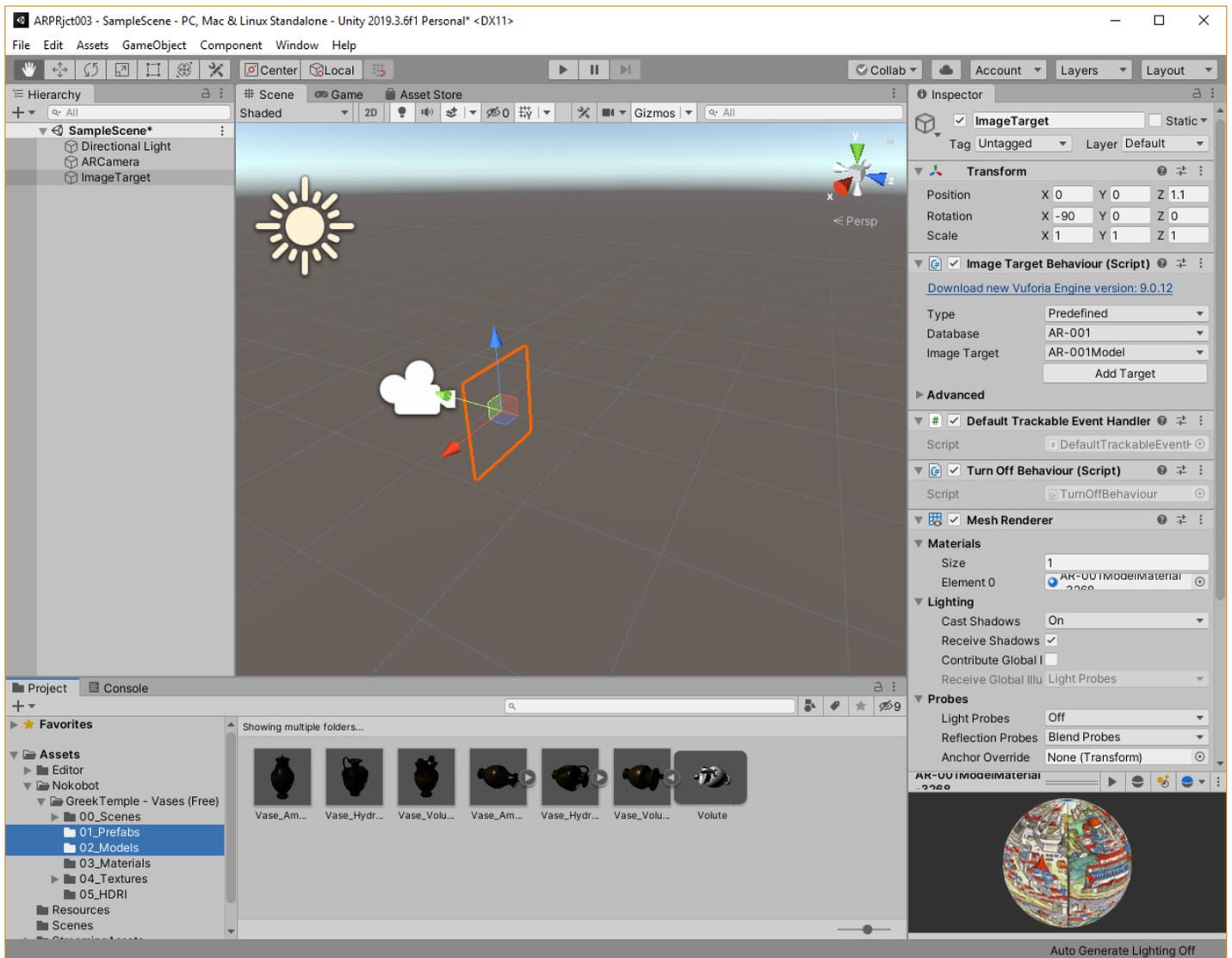


Подтверждаем импорт **3D-моделей** → клавиша **Import**. В результате в области редактора **Unity 3D Project View**, в области **Assets** появляется новый актив с именем **Nokobot**, что соответствует имени корня импортированной структуры. Обратите внимание, что в структуре

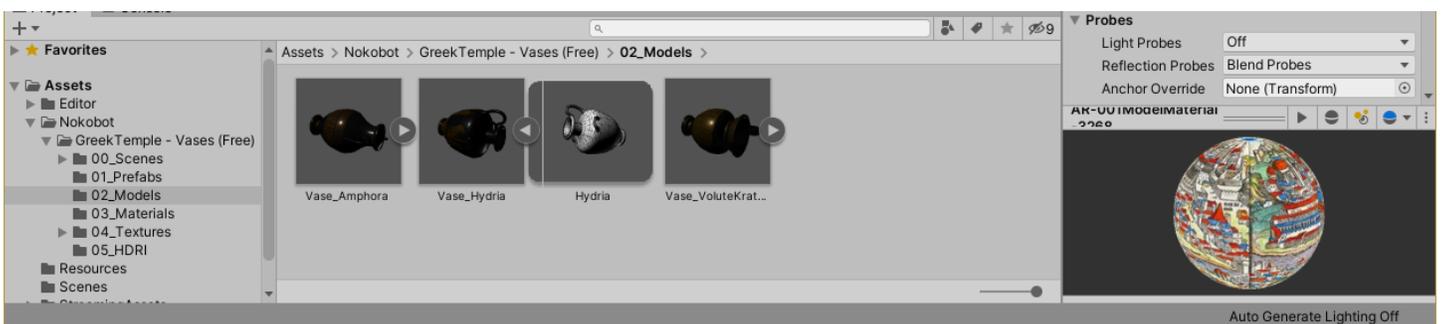
актива содержатся **3D-модели** в одном из допустимых для **Unity** форматов (**.FBX**), особого вида объекты **Unity 3D** под названием **Prefabs** и элементы декоративного оформления поверхностей – текстуры.



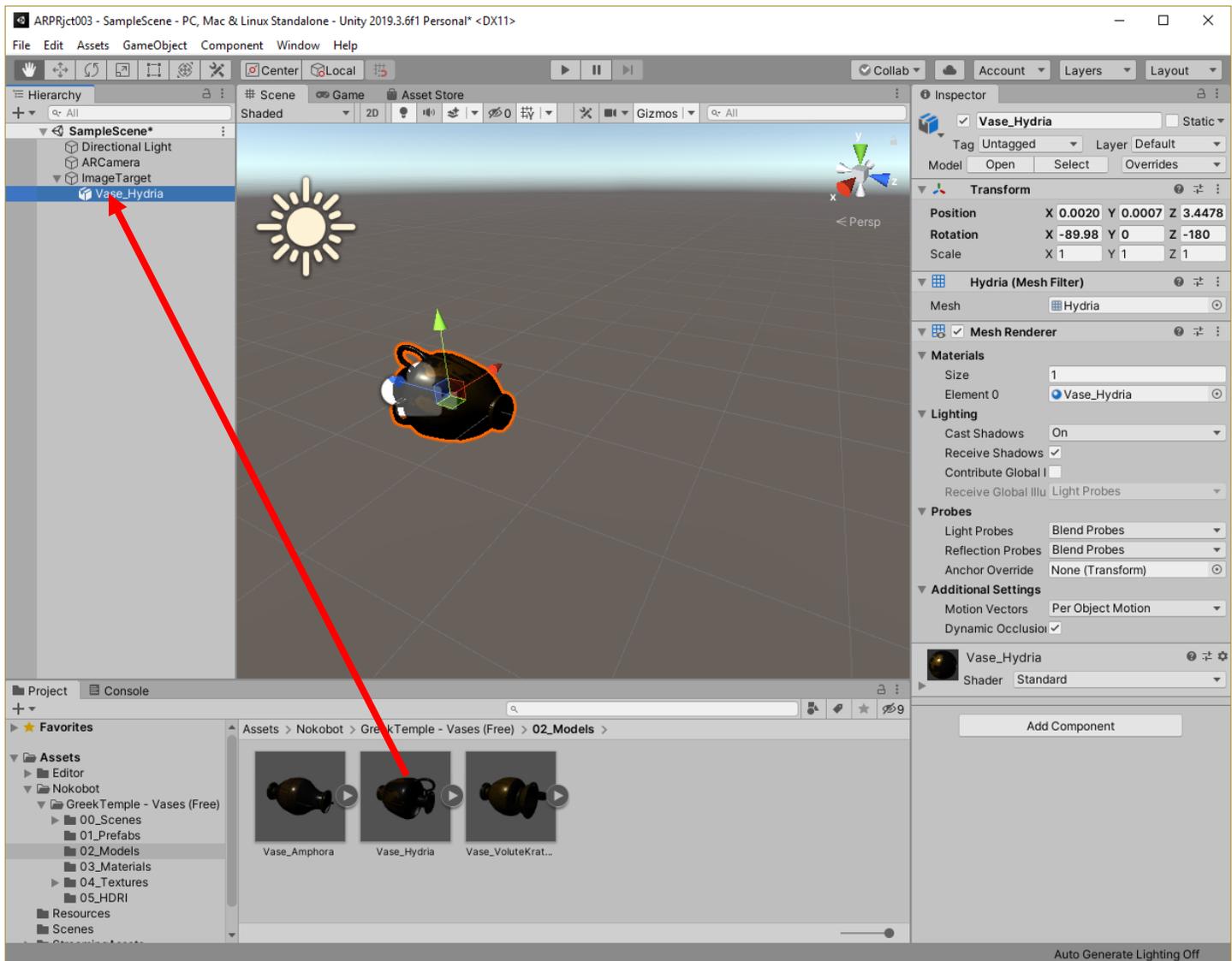
Все готово для размещения **3D-модели** (одной из импортированных в рамках структуры **Nokobot**) в сцене. Возвращаемся в закладку **Scene**:



Теперь связываем одну из нескольких импортированных **3D-моделей (Vase Hydria** в формате **.FBX** в поле **Assets:**)



с зарезервированным таргетом нашего проекта путем перетаскивания (**Drag-n-Drop**) пиктограммы **Vase Hydria** на объект **Image Target** в редакторе сцены (левая часть экрана - **Hierarchy**).

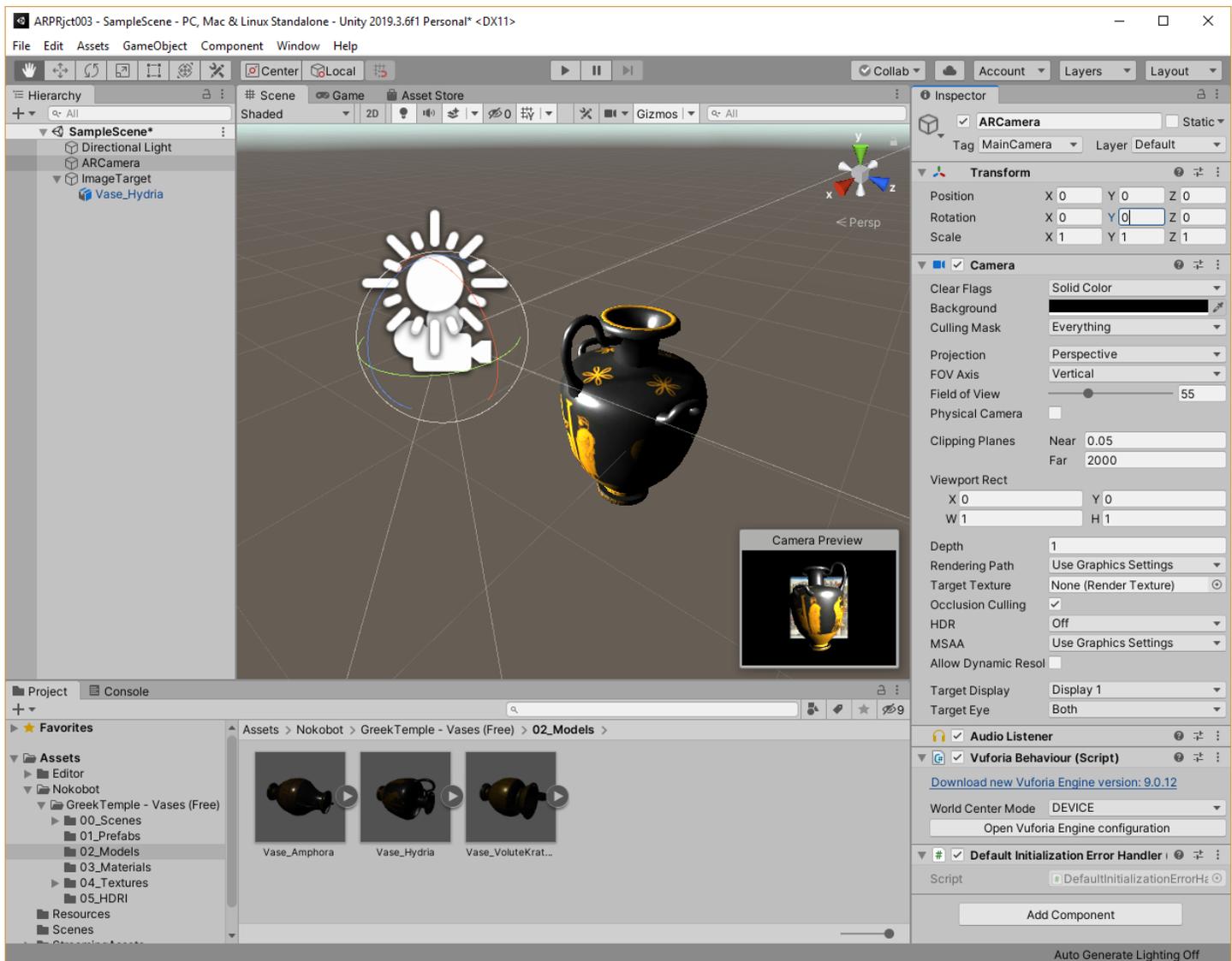


Полученный результат: импортированная **3D-модель** из «магазина» плюс все сопутствующие файлы теперь связаны с таргетом нашего проекта.

Выполняем необходимые или желательные преобразования масштаба и взаимного расположения **3D-модели** и таргета, выбирая в сцене связанную с таргетом **3D-модель** - **Vase Hydría** (обратите внимание, что этот объект относится к типу **Meshes** – что еще раз подчеркивает то обстоятельство, что по умолчанию **Unity 3D** акцептирует поверхностные **3D-модели** в перечисленных в документации форматах: <https://docs.unity3d.com/ru/current/Manual/class-Mesh.html>). Для того, чтобы получить наиболее выгодное расположение **3D-модели** в сцене, воспользуйтесь регулировками в области **Inspector (Position, Rotation, Scale....)** или элементами управления



для каждого из элементов **Hierarchy** Сцены: **Directional Light**, **ARCamera**, **Image Target**, **Vase_Hydría**.



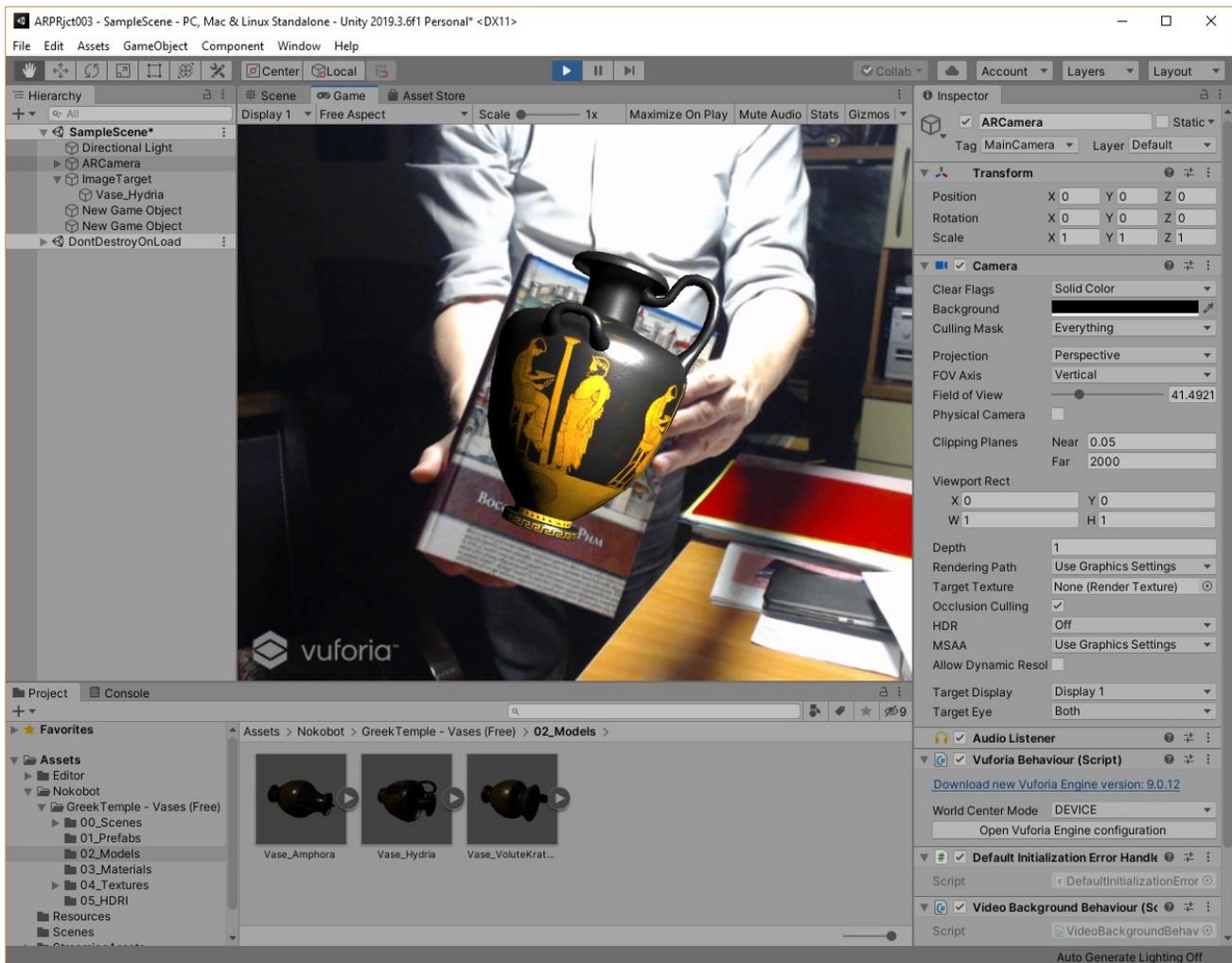
Проверить работоспособность Приложения ДР непосредственно в **Unity 3D** можно на локальной машине, если она снабжена видеокамерой.

Для этого в окне **Unity 3D** необходимо войти в режим **Play** – найти и нажать на клавишу проигрывания («**Play**») в верхней части окна, как показано на рисунке ниже:

Тест: в область просмотра камеры локальной машины помещается твердая копия выбранного для данной сцены таргета:

Тест: по выбранному таргету автоматически прикрепляется объект дополненной реальности. В данном случае – это 3D-Модель:

<https://youtu.be/JoHuOJZhiDQ>



Разработка сцены закончена. Можно сохранить полученный результат – сцену - не выходя из **Unity 3D**. Это может пригодиться в дальнейшем при доработке сцены, при прерывании сеанса работы в **Unity 3D** и т.д. Для сохранения сцены выполнить: **File→Save** или **File→Save as**. Сохранение производится в локальной ФС на вашей локальной машине.

Пункт 7 выполнения ЛР. Создание файла **.apk** для загрузки Приложения ДР на **Android-МУ**.

Все установки, настройки и манипуляции с окнами и функциями **Unity 3D** для создания файла **.apk** идентичны тем, что вы выполняли в ЛР№2, части 1, 2. Не забудьте снабдить свое загружаемое на МУ Приложение иконкой!

Сохраняем собранный файл **.apk** в локальной файловой системе разработчика, загружаем и устанавливаем его на **Android - МУ** любым известным вам способом.

Видео разработанного в данной ЛР Приложения ДР доступно по ссылке:

<https://youtu.be/ChkRJabrpIs>

Разработанное в рамках ЛР № 2, часть 3 AR-Приложение необходимо продемонстрировать преподавателю. Для этого загрузите свой **.apk** на любой файлообменник или доступное облако и пришлите мне ссылку!

Не забудьте про таргет!