

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

پروژه دوره کارشناسی

موضوع:

نرم افزار واقعیت افزوده دانشکده شهید شمس پور

ارائه شده به

گروه کامپیوتر

دانشکده شهید شمس پور

به راهنمایی

استاد

دکتر حمید خورسند رحیم زاده

توسط دانشجویان

ایلیا لسانی بحری

سید علیرضا میرزا قوامی

به شماره دانشجویی

۹۶۱۲۰۰۳۳۱۱۱۰۶۵

۹۶۱۲۱۰۳۳۱۱۱۰۶۴

تاریخ دفاع

۹۸/۰۴/۱۲

چکیده

هدف این پروژه ساخت و توسعه برنامه کاربردی پردازش تصاویر با تکیه بر الگوهای خاص در قالب واقعیت افزوده^۱ و توزیع آن بر روی تلفن های همراه هوشمند بوده است. مطالعه و بررسی زمینه واقعیت های دیجیتال^۲ به رشد روز افزون برنامه های واقعیت مجازی^۳ و واقعیت افزوده اشاره می کند. به کارگیری این فناوری نوین باعث تلفیق دنیای دیجیتال و واقعیت شده و انجام برخی از کارهای ضروری مانند آموزش، تعامل کاربر را اختیار حاصل کرده و انجام امور تعیین شده را تسریع می بخشد.



برنامه کاربردی واقعیت افزوده دانشکده شهید شمس پور در نظر دارد تا دانشجویان را مخاطبان اصلی خود قرار داده و در انجام کارهای اداری مرتبط در آشنایی با دپارتمان های مختلف دانشکده و مراحل نحوه انجام برخی از امور اداری مانند درخواست معافیت تحصیلی و مشخص کردن محل درخواست آن ها یاری کند. این برنامه کاربردی در زمان توسعه بارها در محیط عملیاتی از پیش تعیین شده مورد آزمایش قرار گرفته و برای برخی از دانشجویان به نمایش درآمده است. نظرات مخاطبان حاضر در صحنه بسیار مثبت بوده و اشتیاق آن ها برای استفاده از این نرم افزار را نشان می داد. از همین بازخورد، تصمیم بر آن شده تا در حد امکان از اکثر تلفن های مجهز به سیستم عامل هوشمند پشتیبانی شود.

نام برنامه نیز برای سادگی و انتقال معنای مد نظر توسعه دهندگان، "واقعیت افزوده شمس پور"^۴ اتخاذ شده است.

¹ Augmented Reality

² Digital Realities

³ Virtual Reality

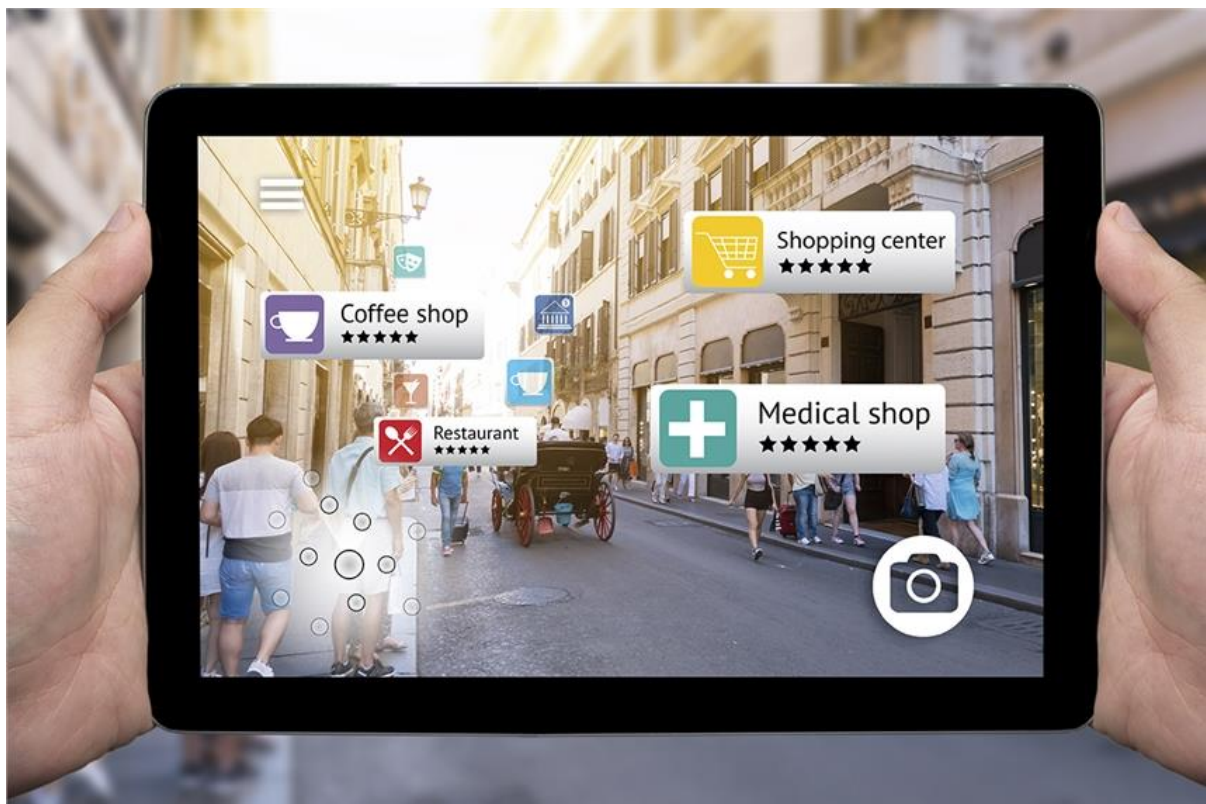
⁴ Shamsipour AR

فهرست

۶	پیشگفتار
۷	اسکلت نرم افزار
۹	عملکرد
۹	قابلیت های کلیدی نرم افزار Adobe XD
۱۰	طراحی اولیه گرافیکی با Adobe XD
۱۱	زبان برنامه نویسی
۱۳	بانک اطلاعات
۲۶	الگوریتم ها
۲۹	نصب و راه اندازی
۳۹	مشخصات سیستم مورد نیاز
۴۰	بانکها و جداول
۴۵	ساختمان داده
۴۶	دستورات برنامه
۵۱	پیشنهاد
۵۱	نتیجه گیری
۵۲	راهنمای کاربری
۵۳	ضمائم
۵۴	منابع و مآخذ

پیشگفتار

انگیزه تیم توسعه دهنده از انتخاب موضوع واقعیت افزوده، آینده روشن آن و تصمیم به فعالیت و توسعه برنامه های بیشتر در این زمینه بوده و بر اساس این علت تصمیم بر آن شد تا پروژه دانشگاه نیز به این موضوع تخصیص گردد. واقعیت افزوده به معنای تلفیق عناصر مجازی و دیجیتالی با دنیای حقیقی، از دید کاربر از طریق یک دستگاه هوشمند است. بدین معنا که کاربر با در دست داشتن دستگاه هوشمند خود و دیدن محیط اطراف خود از طریق دوربین آن، علاوه بر حقایق دنیای پیرامون، عناصر دیجیتالی را مشاهده می کند. همچنین این فناوری در عرصه تبلیغات و تعامل بهتر با کاربر و امور تجاری مطرح می گردد.



به گفته مدیر عامل شرکت اپل آقای تیم کوک:

"از نظر من اهمیت واقعیت افزوده به اندازه گوشی های هوشمند زیاد است. گوشی هوشمند محصولی است که به همه تعلق دارد و نمی توان تصور کرد که آیفون محدود به کشور، موقعیت جغرافیایی یا فرهنگ خاصی باشد. من تصور می کنم واقعیت افزوده به همین اندازه بزرگ است، چراکه این فناوری می تواند قابلیت هایی در اختیارمان قرار دهد که با استفاده از آن ها می توانیم علاوه بر جذابیت زندگی، کیفیت زندگی خود را نیز افزایش دهیم."

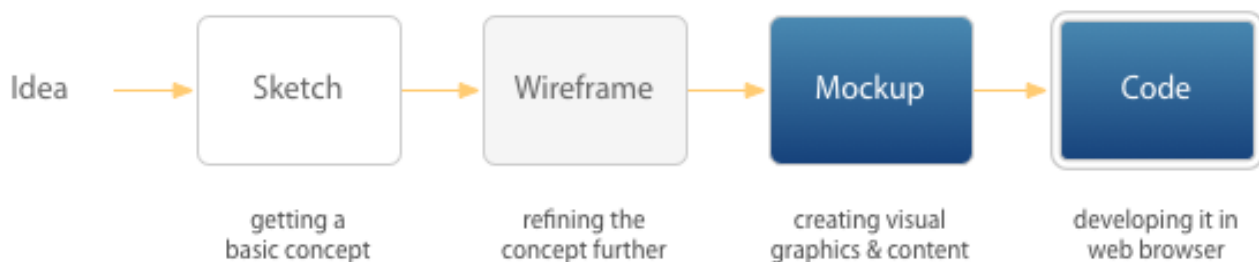
اسکلت نرم افزار

پیشنهاد یک ایده برای نرم افزار کار آسانی است اما درک نحوه شکل گرفتن ایده در قالب واسط کاربری کار سخت تری می باشد. طراحی اولیه^۵ ظاهر نرم افزار بر روی کاغذ شکل گرفته و مورد بررسی قرار می گیرد. چندین طرح پیشنهاد داده شده و مناسب ترین آن برای رفتن به مرحله بعد انتخاب می شود. در اکثر حرفه ها که شامل خلاقیت و توسعه می باشند، این نوع طراحی کاربرد دارد.

هر قدر هم که طراح نرم افزار ماهر باشد، پیاده سازی مستقیم ایده بدون هیچ پیش زمینه ای و مشخص کردن جزئیات تقریباً کار غیرممکنی است. هر ایده ای نیاز به ترجمه به واسط کاربری دارد و یک فرآیند محسوب می شود. فقط گفتن برنامه ای که کار الف و ب را انجام می دهد کافی نیست. برای انجام کارهای الف و ب برای هر کدام از آن ها نیاز به طراحی یک صفحه جداگانه یا تخصیص بخشی از صفحه نیاز است.

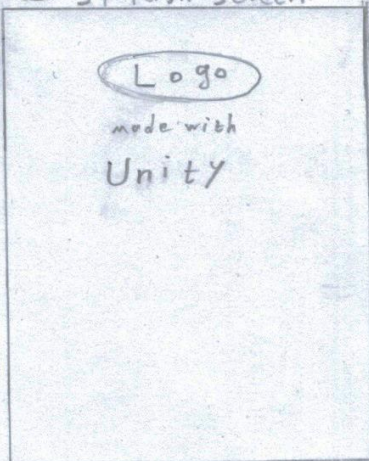
«طرح ریزی اولیه یک نرم افزار امکان بصری سازی تعامل صفحه به صفحه را ممکن می سازد»

اما این نکته قابل ذکر است که یک قاعده کلی و استاندارد برای طی کردن فرآیند طراحی وجود ندارد و این عمل به سلیقه طراح صورت می گیرد. از دیگر مزایای این طرح، پیشبرد سریعتر کارها و امکان ساده تر کردن مراحل بعدی پیاده سازی است.

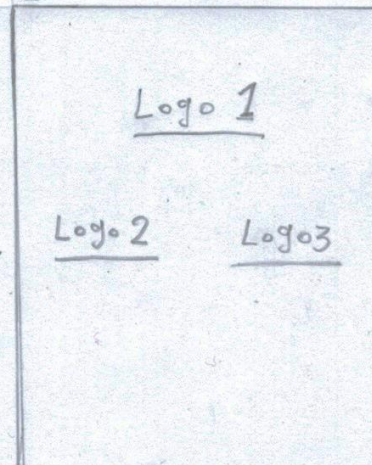


⁵ Sketch

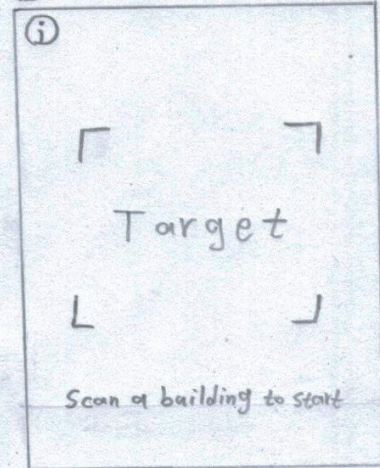
① Splash Screen



② Logos



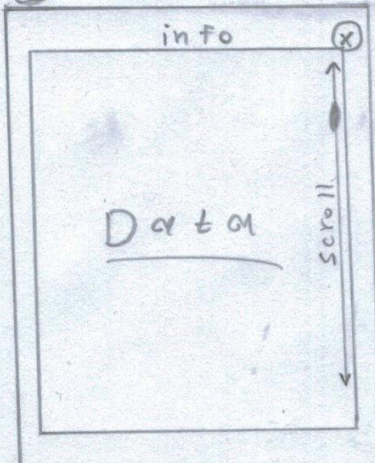
③ Main Scene



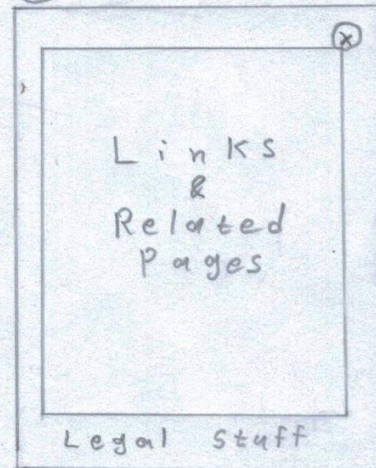
④ Trigger(s)



⑤ About



⑥ About Developers



① symbol provides information about the developers

عملکرد

در مرحله بعد اسکلت کلی نرم افزار بر طبق طراحی اولیه تولید می شود.

تجربه کاربری شامل رفتارها، نگرش ها و احساسات یک کاربر درباره استفاده از یک محصول است و به عبارتی خاطره ای است که در ذهن کاربر نقش می بندد. بنابراین در فرآیند ساخت و توسعه محصولات از اهمیت ویژه ای برخوردار است. نرم افزار Adobe XD راه حل کامل برای طراحی، نمونه سازی و به اشتراک گذاری تجربه کاربری و رابط کاربری برای انواع خدمات و محصولات نرم افزاری است. این نرم افزار به منظور برآورده سازی نیازهای طراحان ارائه شده است تا به وسیله ابزارهای بصری که دقت و عملکرد بالایی دارند، انجام کار طراحان را ساده تر و پیشبرد نمونه سازی محصولات را سریعتر می کند. استفاده از قابلیت های این نرم افزار باعث شده تا به طور قابل توجهی در زمان مورد نیاز برای پیاده سازی نرم افزار صرفه جویی و طرح های انعطاف پذیری را از طرح های کلی و ساده تا نمونه های کاملاً تعاملی تنها در مدت زمان کوتاهی نمونه سازی شود.

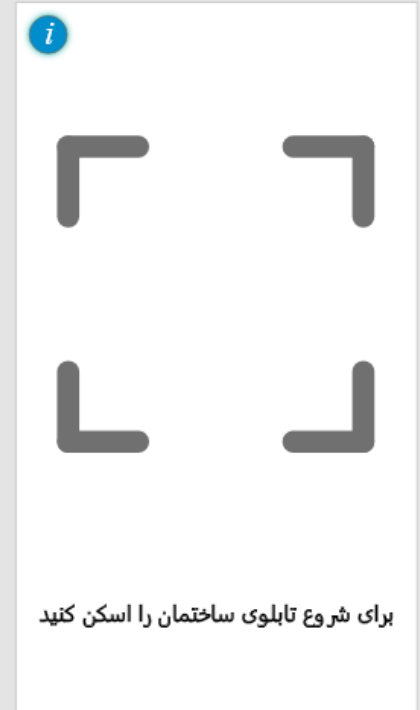


قابلیت های کلیدی نرم افزار Adobe XD:

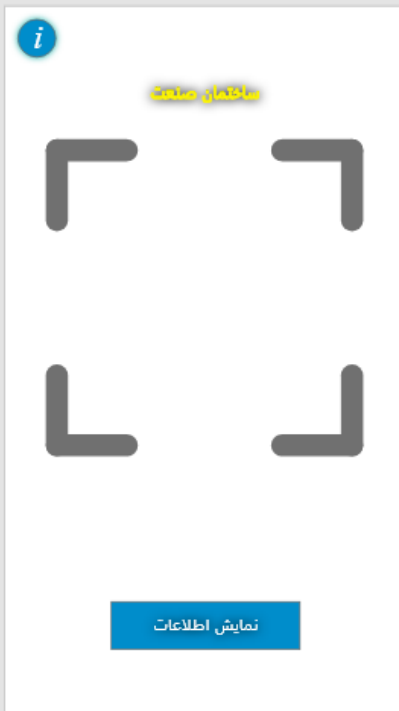
- آماده سازی طرح اولیه و نمونه سازی تعاملی
- دستیاری قدرتمند برای طراحان UX/UI
- انتشار نمونه ها در راستای گرفتن بازخورد
- طراحی و نمونه سازی رابط کاربری و بررسی تجربه کاربری
- سازگاری کامل با نرم افزارهایی نظیر Photoshop و Illustrator
- ترکیب سطوح طراحی با یکدیگر برای تجسم راهبرد در برنامه های بزرگ

طراحی اولیه گرافیکی با Adobe XD

Shamsipour AR — Edited ▾



Android Mobile – 4



Android Mobile – 5

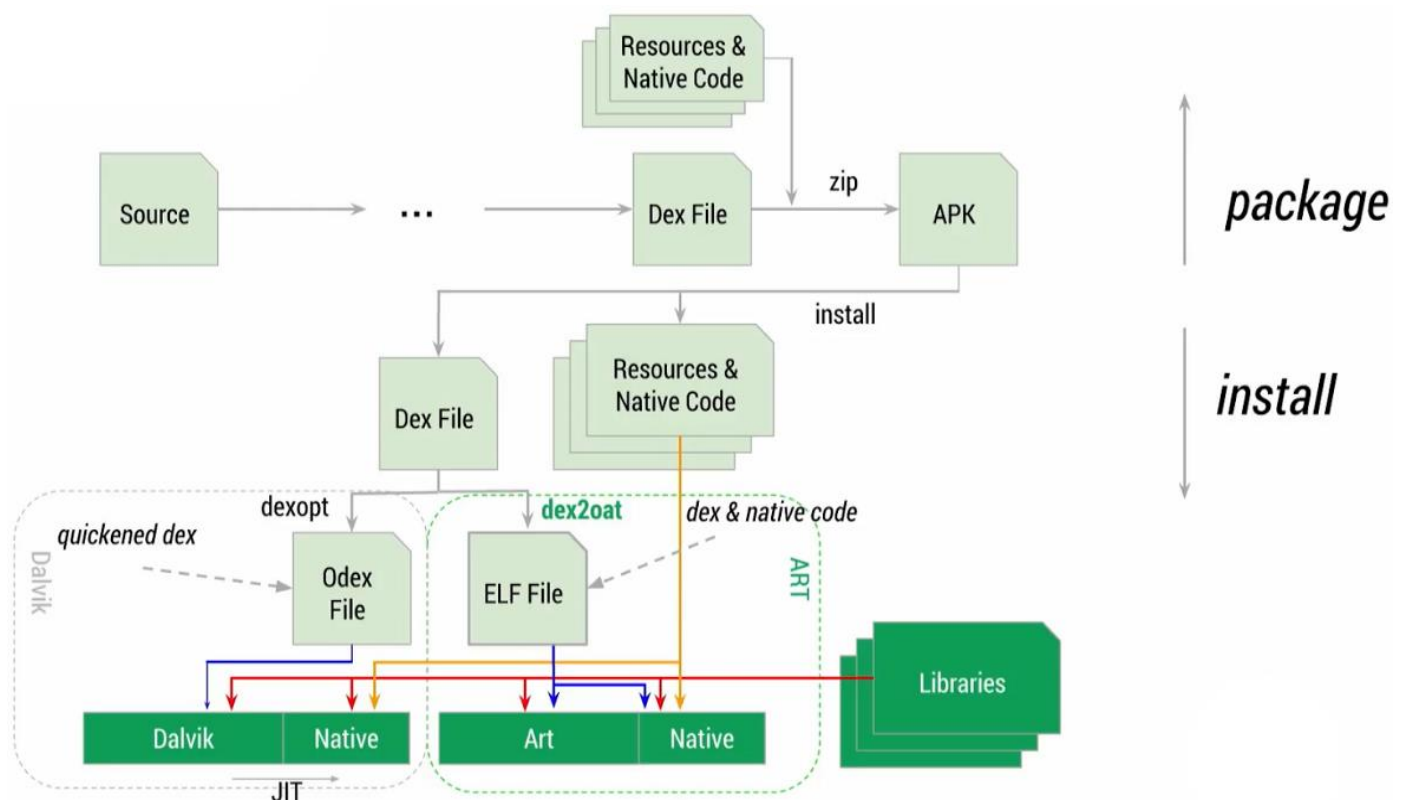


Android Mobile – 6



زبان برنامه نویسی

نرم افزار مذکور از زبان برنامه نویسی سی شارپ (Mono C#) به همراه فریم ورک واقعیت افزوده Vuforia استفاده می کند. کد های برنامه به ترتیب برای پلتفرم های Android و iOS به کد های Java و C++ تبدیل می شود. مراحل کامپایل شدن کد سی شارپ به جاوا و سی پلاس پلاس متفاوت است. در کامپایل اندروید، ابتدا یک پروژه از نوع استودیو اندروید (Android Studio) گرفته می شود. این پروژه شامل تمام ابزار توسعه اندروید (Java ADT)، سیستم ساخت Gradle و کتابخانه های مورد نیاز برای اجرای پروژه در پلتفرم اندروید می باشد. برنامه های اجرایی اندروید قبل از تبدیل شدن به کد اجرایی ماشین مجازی جاوا، اول به Bytecode و سپس به قالب ELF و ODEX برای تولید کد ماشین کامپایل می شوند. ماشین مجازی اندروید Dalvik و ابزار زمان اجرای آن ART نام دارد و برنامه های اجرایی آن ملقب به DEX می باشد. عملکرد Dalvik به صورت JIT⁶ و ART به صورت AOT⁷ است. فایل های نصبی اندروید DEX های فشرده ای هستند که پسوند APK به آن ها تعلق می گیرد.

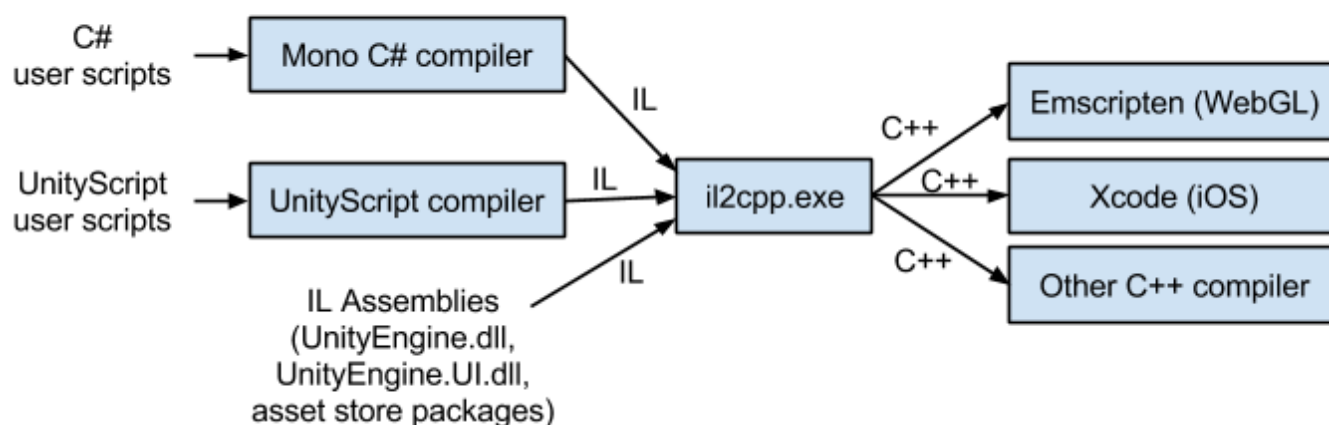


⁶ Just-in-Time compilation

⁷ Ahead-of-Time compilation

تبدیل کد سی شارپ به شکل اجرایی دستگاه های اپل فرآیند متفاوتی است. کامپایلر متن باز سی شارپ دستورات برنامه را به یک زبان میانی به نام IL⁸ تبدیل می کند. زبان میانی توسط برنامه ای به کد های C++ تبدیل شده و می تواند در محیط های توسعه مختلف به کار رود. محیط توسعه دستگاه های iOS، برنامه XCode می باشد که تمام قابلیت های مورد نیاز مانند Loader و Linker و مخصوصا Optimizer را دارا است. XCode نقش تولید برنامه اجرایی برای iOS را با کامپایل کردن کد ها به قالب IPA ایفا می کند.

فرآیند تولید خروجی برای دستگاه های میکروسافت نیز این چنین است. تنها تفاوت آن، استفاده از کامپایلر متفاوت برای C++ و استفاده از فریم ورک NET. می باشد. برخلاف فرآیند قبلی، Visual Studio موظف است تا خروجی مد نظر بر اساس معماری پردازنده های ARM_64، ARM، x86 و x64 را در قالب های APPX و یا EXE تولید کند. نیاز به ذکر است که تولید خروجی APPX برای گوشی های هوشمند مجهز به سیستم عامل Windows 10 بدون در اختیار داشتن مجوز توسعه دهندگان شرکت میکروسافت مقدور نیست.




⁸ Intermediate-Language

بانک اطلاعات

ساختار بانک اطلاعاتی واقعیت افزوده کمی با پایگاه داده های دیگر متفاوت است. ابتدا باید تمام تصاویر و اهداف مد نظر برای اسکن کردن بر روی سایت Vuforia برای پردازش بارگذاری شوند. برای بارگذاری نیاز به تهیه مجوز توسعه دهنده Vuforia است که از طریق License Manager به آدرس <https://developer.vuforia.com> امکان پذیر است.

Add Target

<input type="checkbox"/>	Target Name	Type	Rating
<input type="checkbox"/>	 ساختمان فرهنگ	Single Image	★★★★★
<input type="checkbox"/>	 ساختمان جدید آموزش	Single Image	★★★★★
<input type="checkbox"/>	 انتشارات	Single Image	★★★★★
<input type="checkbox"/>	 ساختمان فنی	Single Image	★★★★★
<input type="checkbox"/>	 سالن ورزشی	Single Image	★★★★★
<input type="checkbox"/>	 سلف سرویس دانشجویان	Single Image	★★★★★
<input type="checkbox"/>	 تغذیه و سلامت	Single Image	★★★★★
<input type="checkbox"/>	 ساختمان صنعت	Single Image	★★★★★
<input type="checkbox"/>	 ساختمان اندیشه	Single Image	★★★★★
<input type="checkbox"/>	 ساختمان قدیم آموزش	Single Image	★★★★★

تصاویر قبل از پردازش نیاز به اطلاعات اولیه دارند.

Add Target

Type:



Single Image



Cuboid



Cylinder



3D Object

File:

Choose File

Browse...

.jpg or .png (max file 2mb)

Width:

Enter the width of your target in scene units. The size of the target should be on the same scale as your augmented virtual content. Vuforia uses meters as the default unit scale. The target's height will be calculated when you upload your image.

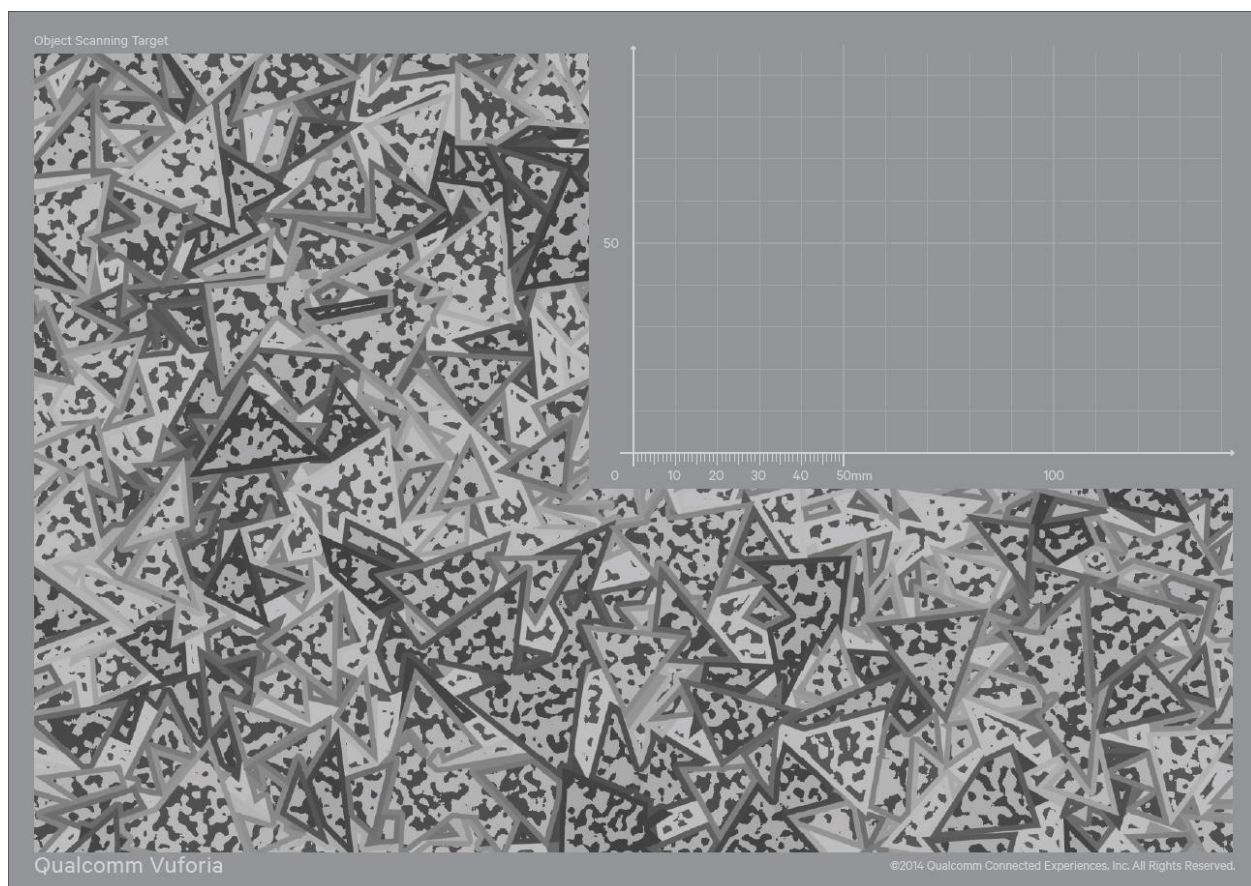
Name:

Name must be unique to a database. When a target is detected in your application, this will be reported in the API.

این اطلاعات شامل نام تصویر در پایگاه داده، طول تصویر (واحد متر) برای تسهیل کار دوربین ها، و نوع تصویر مد نظر می باشد.

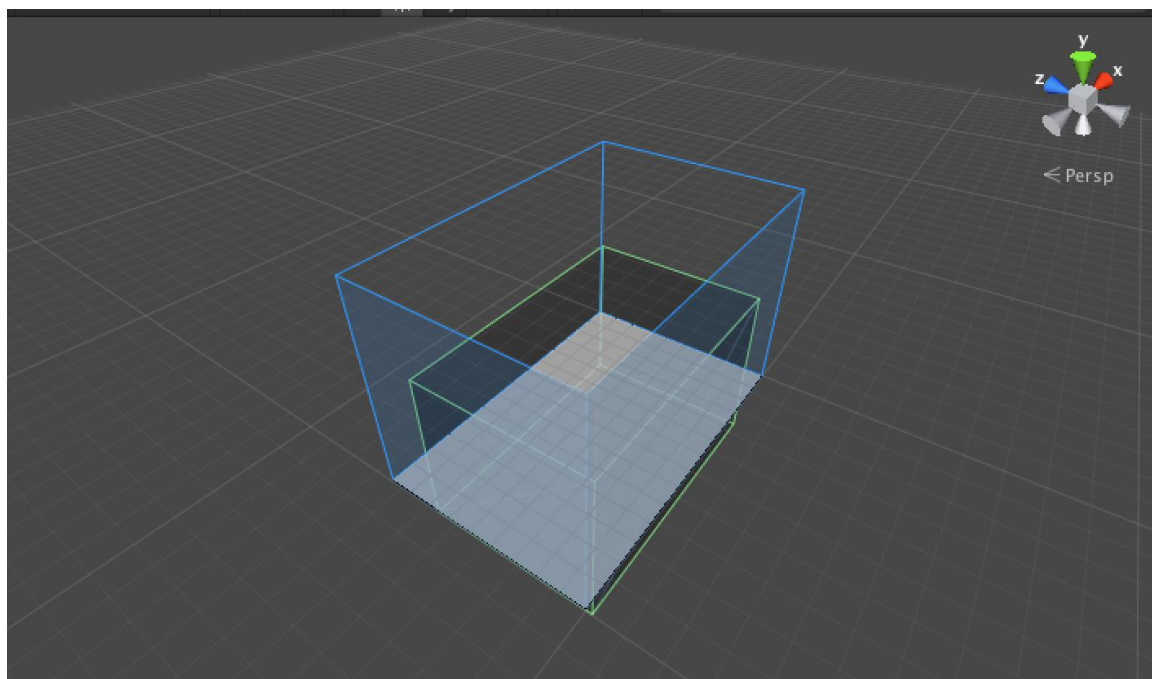
چهار نوع تصویر در پایگاه داده می توانند قرار گیرند. تصاویر Single Image برای اهداف واقعیت افزوده دو بعدی بدون در نظر گرفتن عمق تصویر به کار می روند. Cuboid برای اهداف سه بعدی که از ضلع های مختلف اشیاء مستطیل مانند تصویر برداری شده است، به کار می رود. Cylinder نیز همانند Cuboid ساختاری مشابه را دارا است و برای اشیاء دایره مانند و بدون ضلع های مماس به کار می رود. 3D Object جدیدترین نوع هدف برای واقعیت افزوده Vuforia می باشد. برای تهیه این نوع اهداف باید برنامه Vuforia Object Scanner بر روی دستگاه های پشتیبانی شده⁹ نصب کنید.

قبل از اسکن نیاز به چاپ یک برگه تیره مخصوص اسکن اشیاء در Vuforia است.



⁹ <https://library.vuforia.com/content/vuforia-library/en/articles/Solution/Vuforia-Supported-Versions.html>

این برگه به اسکن هدف در اندازه واقعی کمک می کند و در موتورهای گرافیکی مانند Unity3D در اندازه واقعی ظاهر می شود.



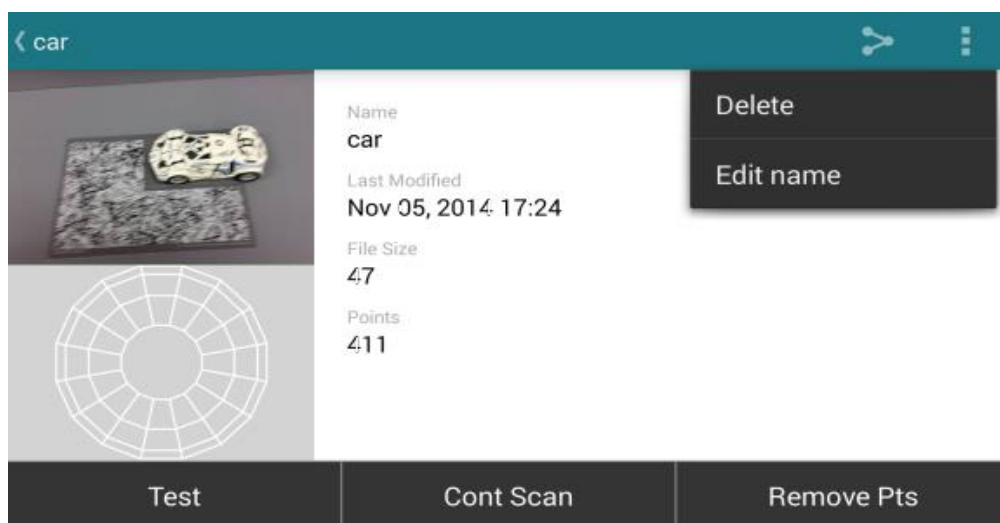
هدف مورد نظر باید در گوشه بالا سمت راست کاغذ قرار گیرد و با نرم افزار اسکن شود.



برای اسکن کامل و سه بعدی هدف، باید از تمام جهات با دوربین اسکن و تمام خانه های سفید رنگ، سبز شوند.

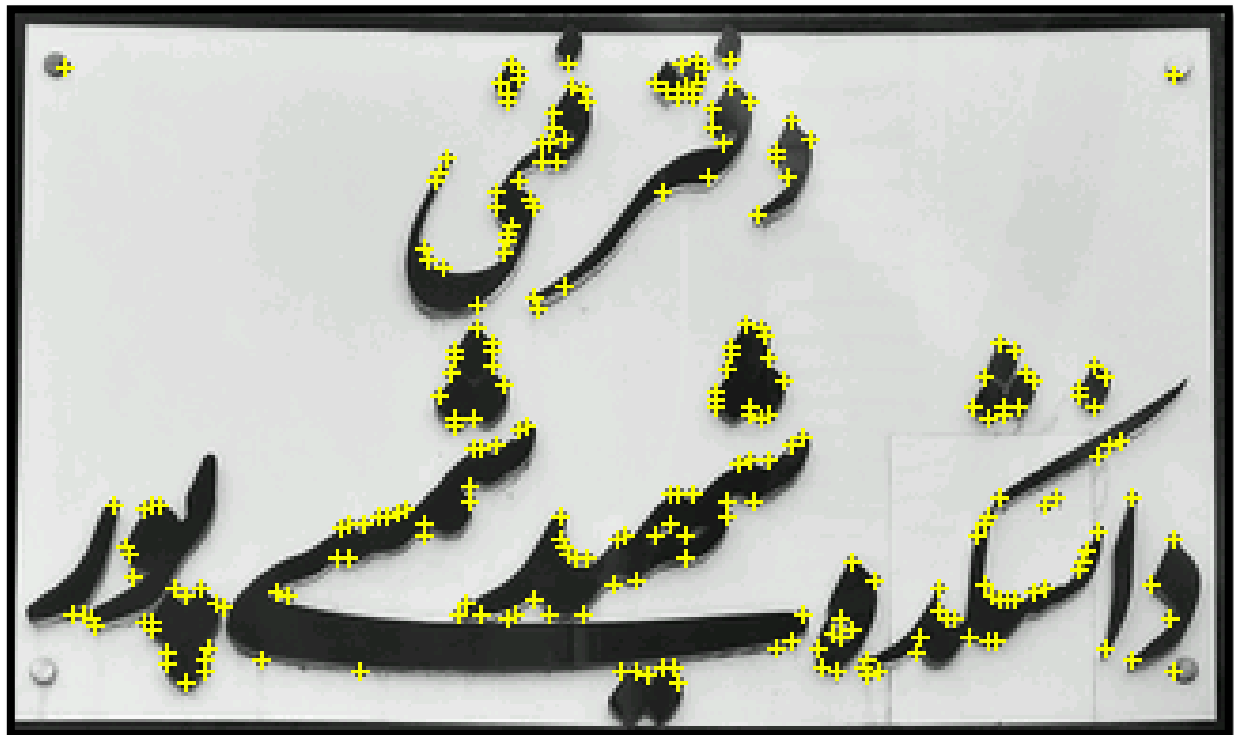


پس از سبز شدن تمامی خانه ها، اسکن کامل شده و می توان با تغییر نام، فایل را ذخیره کرد. پسوند فایل ذخیره شده OD¹⁰ بوده و برای پایگاه داده سایت Vuforia قابل تشخیص است. هر چقدر تعداد نقاط سبز رنگ ریز در هدف اسکن شده بیشتر باشد، هدف جزئیات بهتری داشته و سریعتر توسط واقعیت افزوده پردازش می شود.



¹⁰ Object Data

نام مؤلفه های پایگاه داده نباید تکراری باشد و بهتر است از نام هایی استفاده شود که هدف را خوب توضیح دهد. این نام ها قابل تغییر هستند و در صورت استفاده صریح از نام آن ها درون کد برنامه، ممکن است برنامه با خطا مواجه شود. پس از بارگذاری تمامی تصاویر به پایگاه داده، هر کدام از آن ها به صورت جداگانه پردازش و به آن ها امتیازی تعلق می گیرد.



هر چقدر تعداد نقاط زرد رنگ در بخش ویژگی های پایگاه داده بیشتر باشد، امتیاز تصویر بالاتر و تشخیص آن توسط دوربین گوشی های هوشمند راحت تر می باشد. امتیاز تصویر مذکور پنج ستاره می باشد و به معنای آن است که این تصویر به سرعت تشخیص داده می شود.

Type: Single Image

Status: Active

Target ID: 4947b4fd742f4593866c9c703fd7bfd9

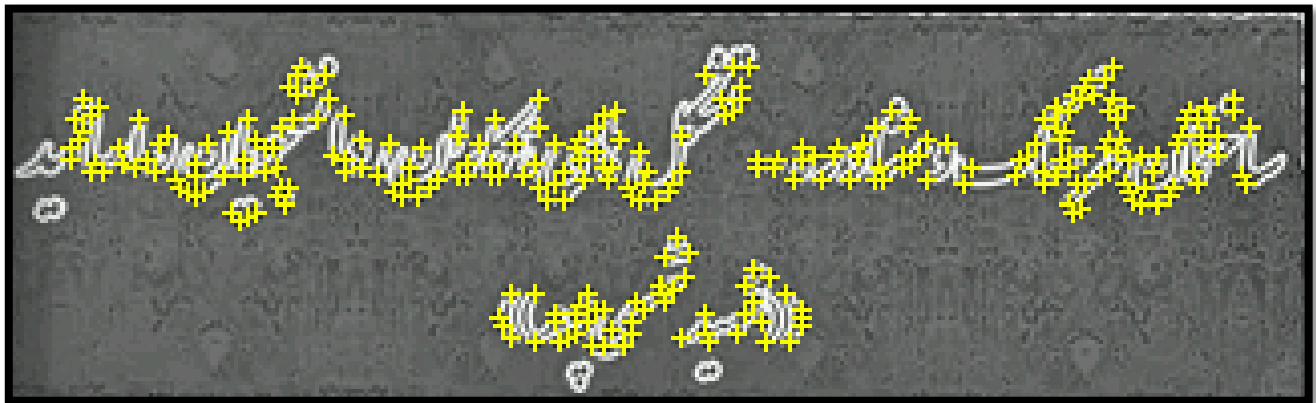
Augmentable: ★★★★★

Added: Apr 6, 2019 20:33

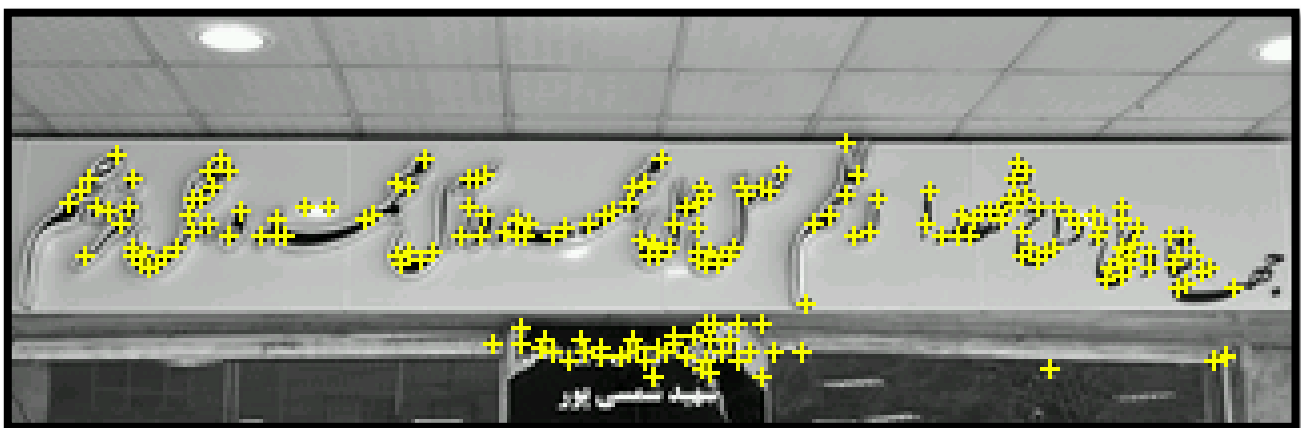
Modified: Apr 6, 2019 20:33

تصاویر پردازش شده در پایگاه داده به این شکل می باشند:

ساختمان فرهنگ

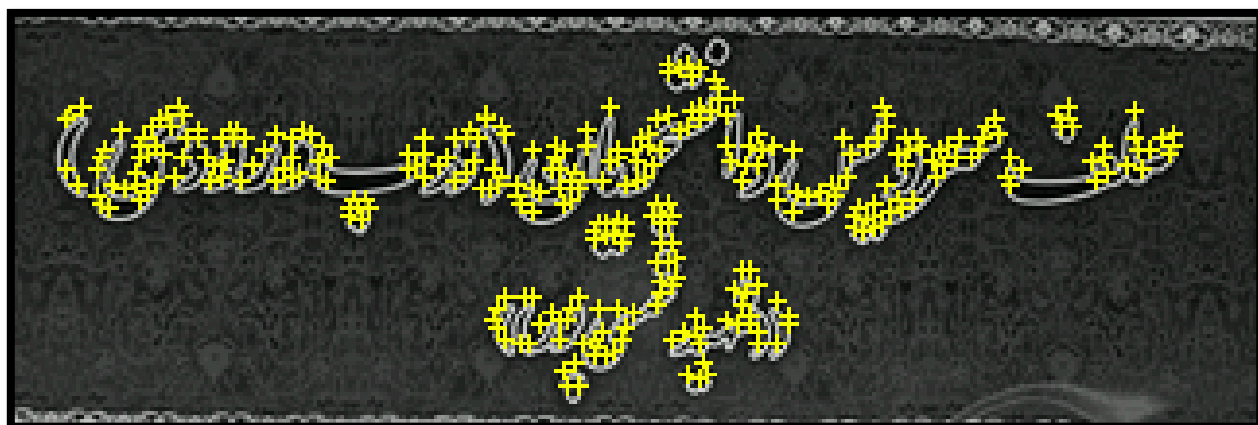


ساختمان جدید آموزش

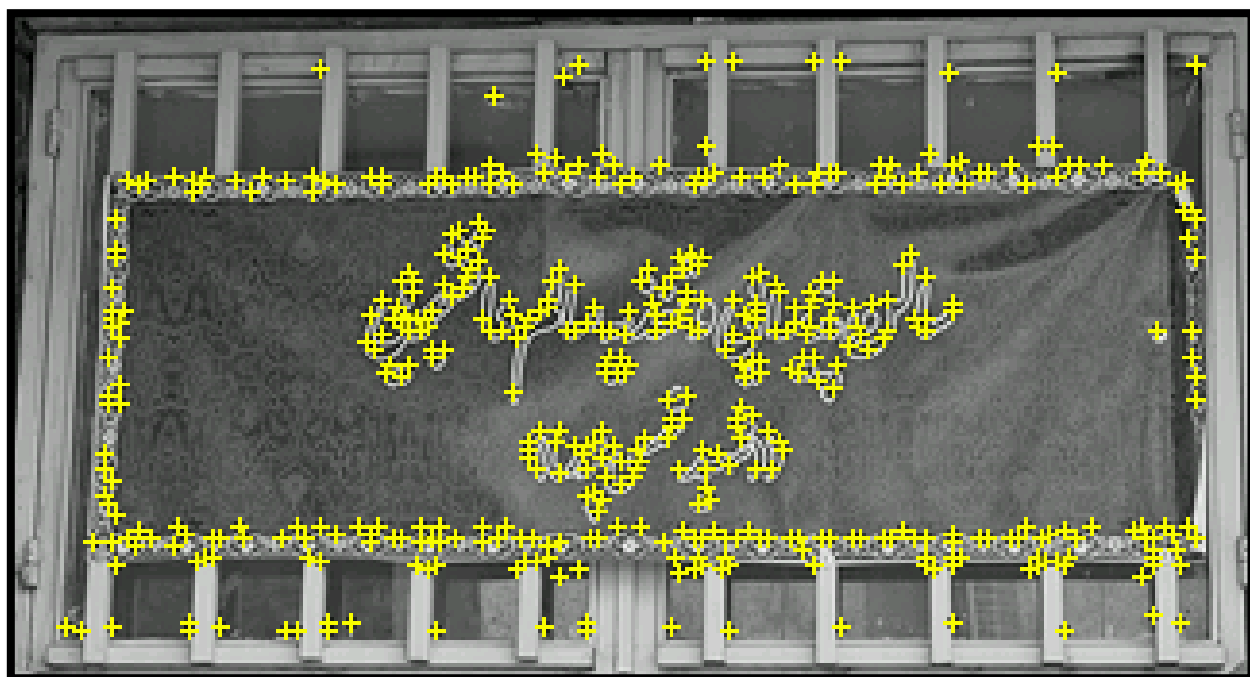




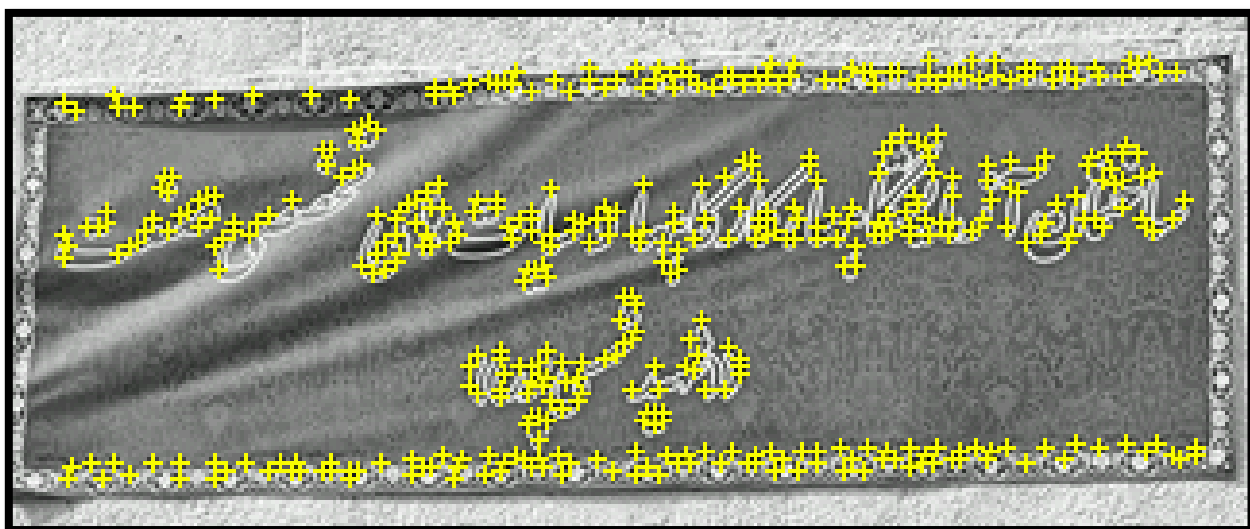
سلف سرویس دانشجویان



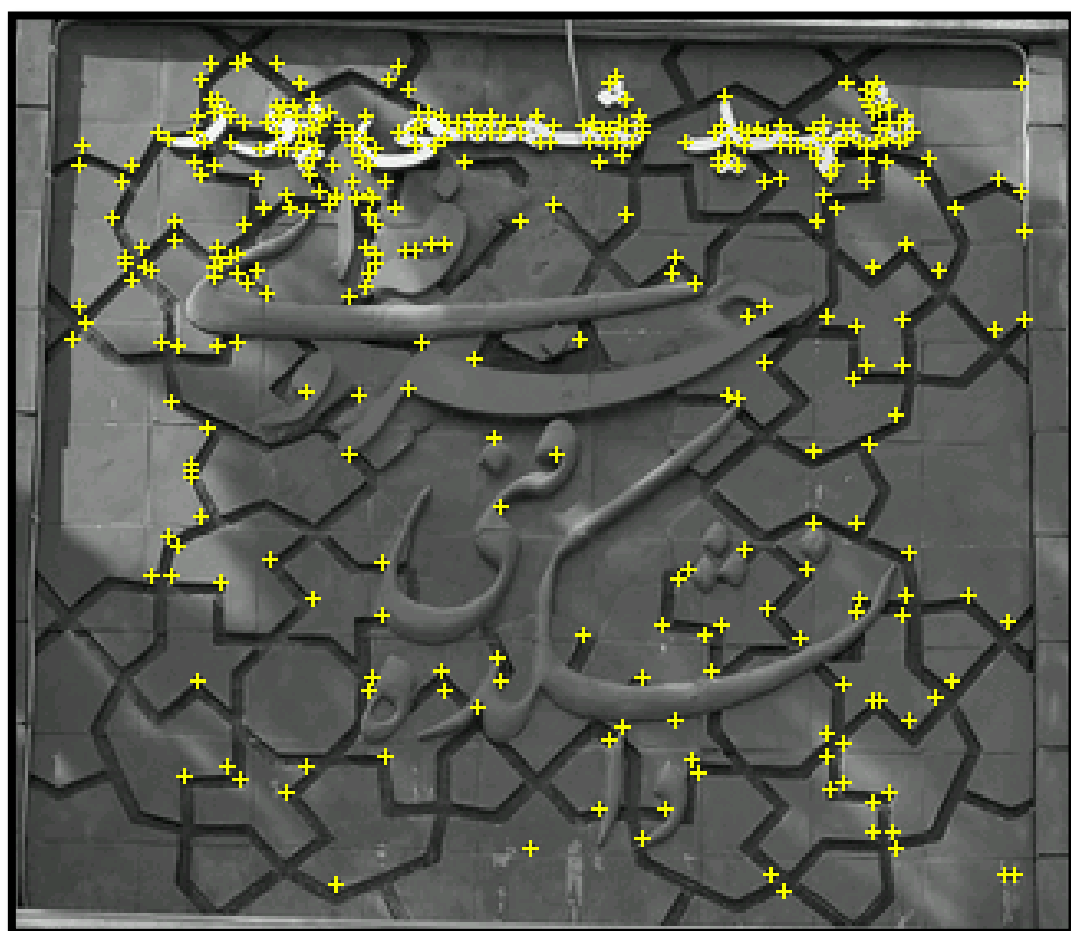
سالن پذیرایی دانشجویان

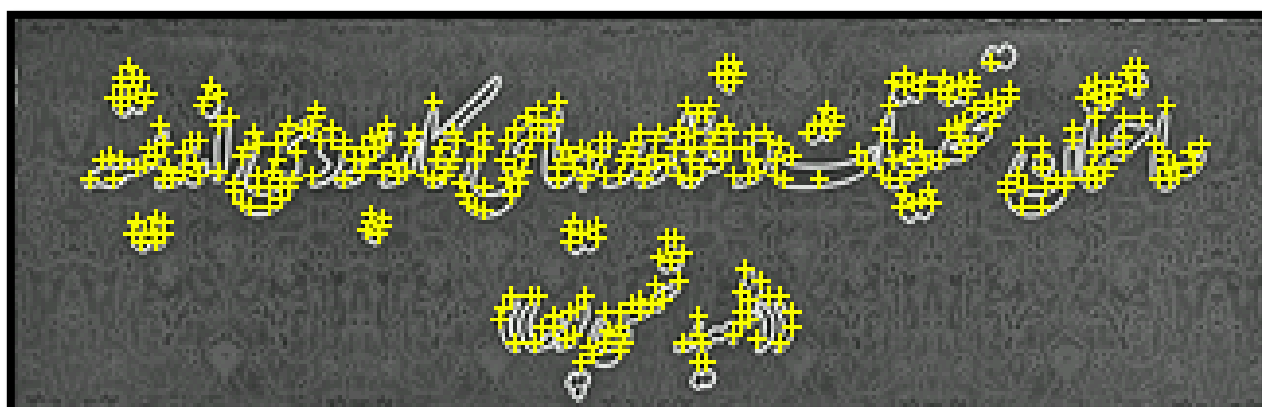


ساختمان صنعت



ساختمان قدیم





پس از اتمام پردازش، گزینه دانلود برای دریافت بانک تولید شده در پایگاه داده را زده و آن را دریافت می کنیم.

Download Database

11 of 11 active targets will be downloaded

Name:

Shamsipour_AR_R_DB

Select a development platform:

☐ Android Studio, Xcode or Visual Studio

☒ Unity Editor

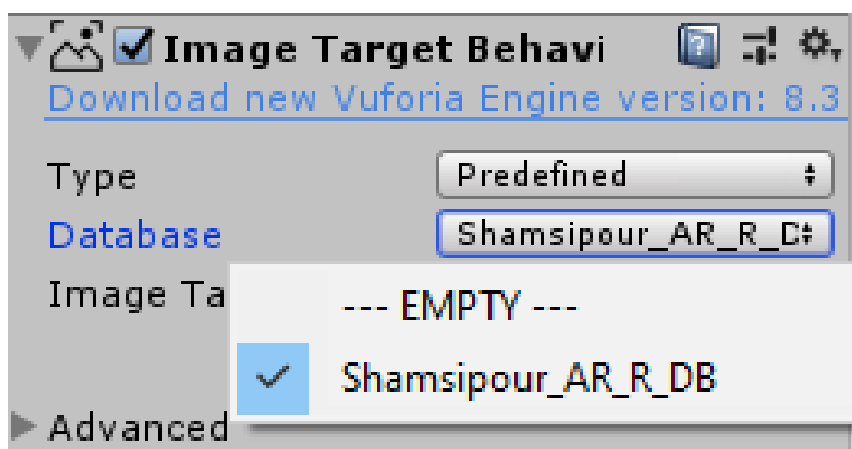
Cancel

Download

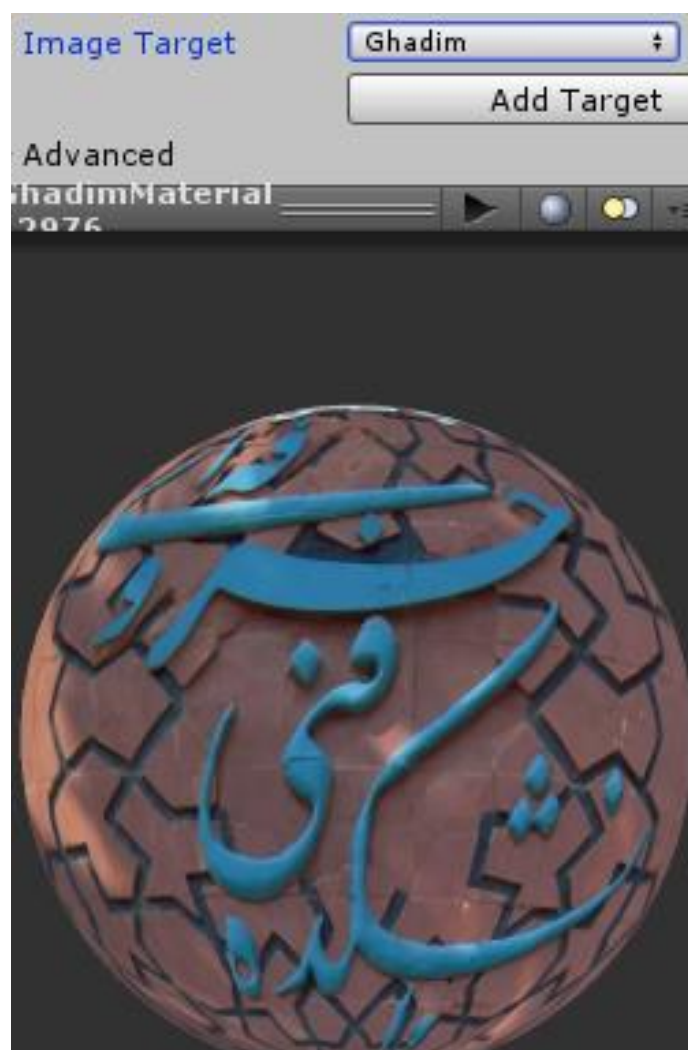
دو نوع بانک جهت استفاده وجود دارد. نوع اول برای استفاده در محیط های توسعه (Integrated Development Environment) به صورت مجزا با پسوند های dat و xml و نوع دوم برای استفاده در موتور Unity3D که امکان تولید یک کد منبع Cross-Platform برای دستگاه های مختلف با پسوند unitypackage را فراهم می سازد.

پسوند unitypackage مختص استفاده در برنامه Unity3D بوده و تنها کافی است به یک پروژه Import شود. تصاویر داخل بانک به Asset های پروژه افزوده می شوند و می توانند با توابع ابزار توسعه Vuforia به کار روند.

سپس پس از import کردن بانک، در بخش Database برنامه باید نام آن مشخص گردد.

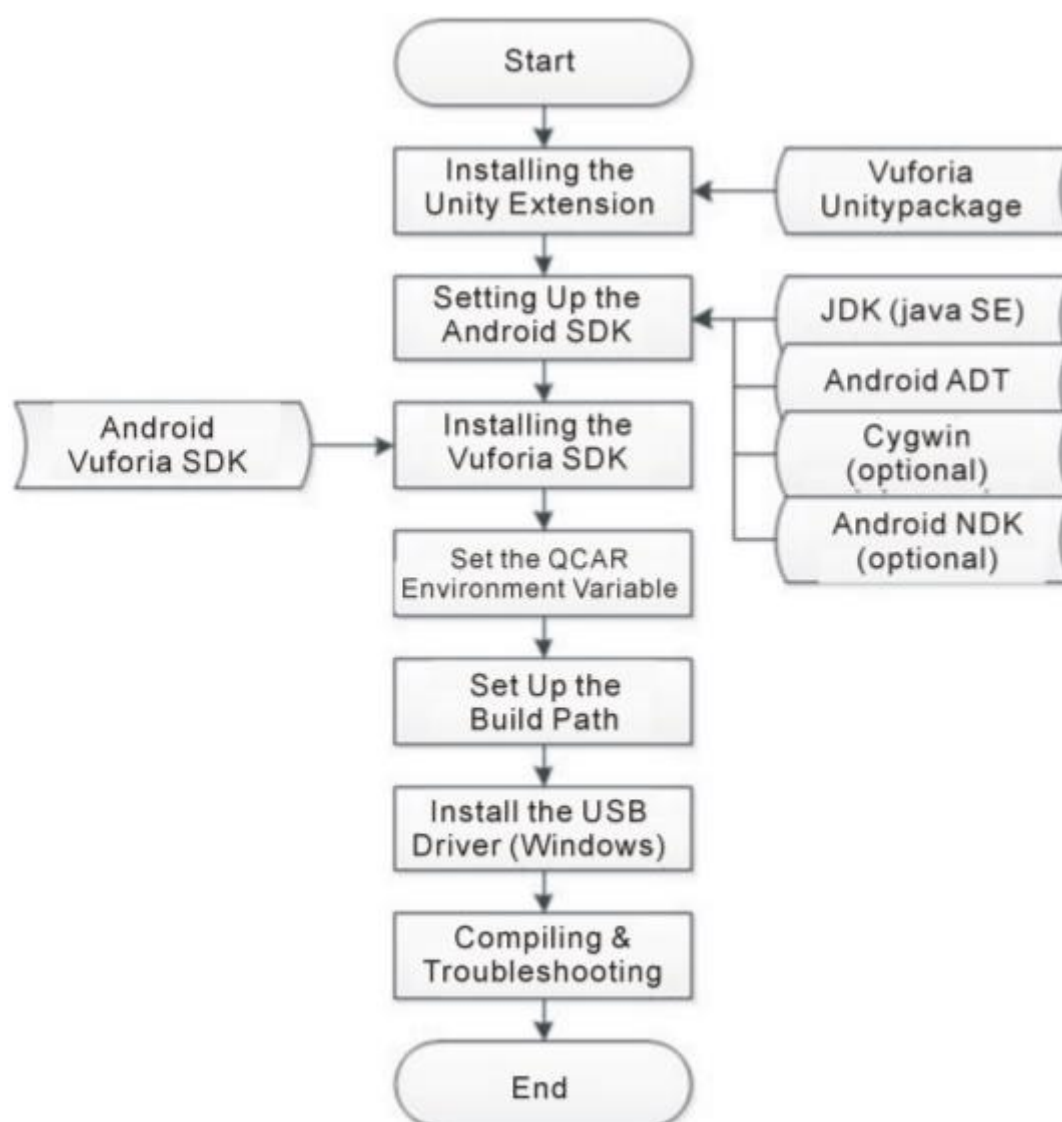


تصاویر از Image Target قابل انتخاب هستند. این تصاویر ساختاری مشابه به Material در Unity3D دارند.



الگوریتم ها

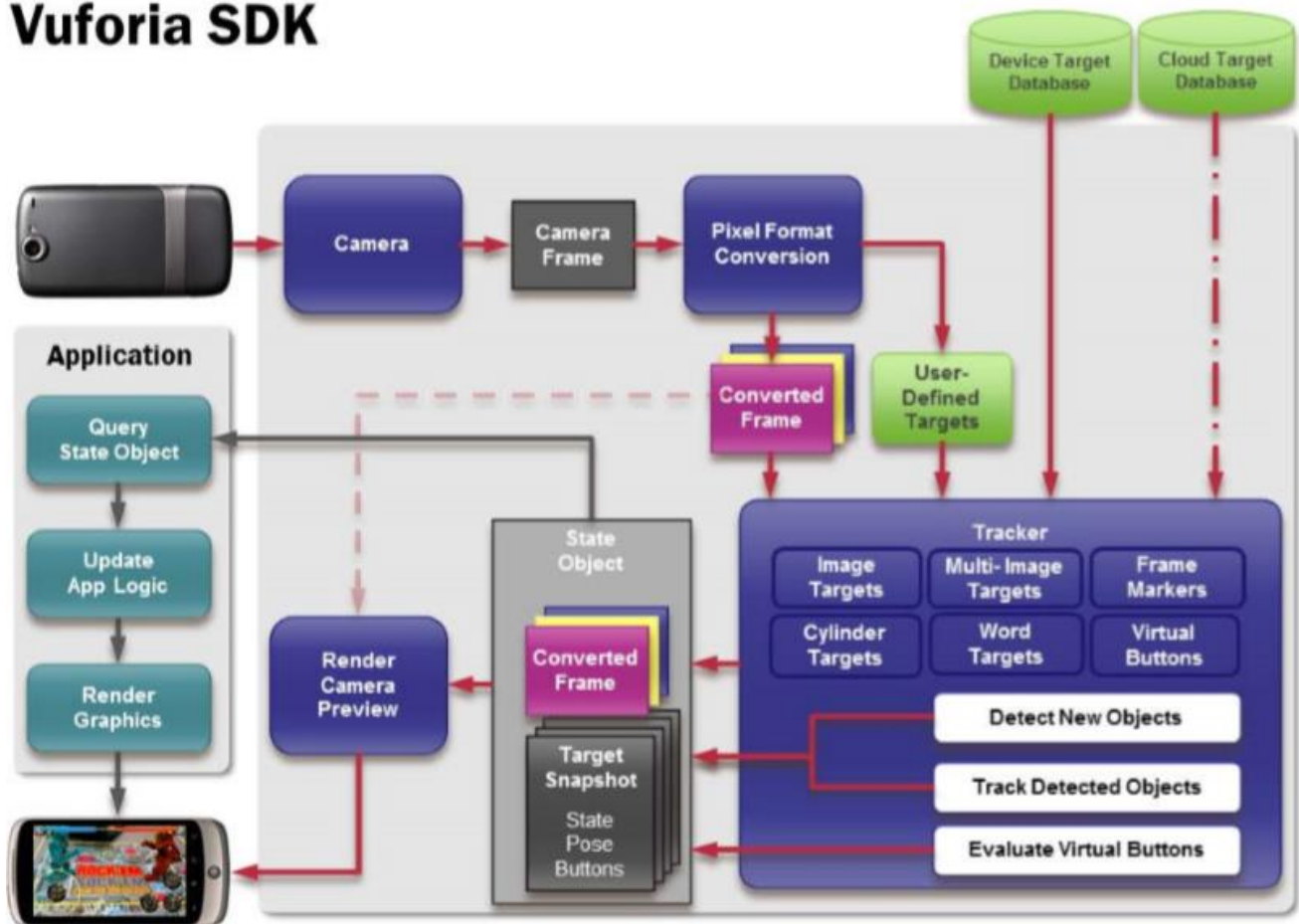
مراحل اولیه انجام کار بدین صورت می باشد. موتور Unity3D به همراه Vuforia نصب می شوند. تمام ابزار مورد نیاز برای گرفتن خروجی اندروید شامل SDK (کد Java SE و ADT) و NDK (کد C++ و Cygwin) نصب و در External Tools موتور Unity3D تعریف می شوند. حال واقعیت افزوده دستگاه های QCAR^{۱۱} اندروید آماده می باشد و جهت آزمایش مستقیم بر روی دستگاه با اتصال سیم، دریافت درایور متناسب USB با توجه به برند سازنده موبایل، باید توسط کاربر انجام گردد.



¹¹ Qualcomm Augmented Reality

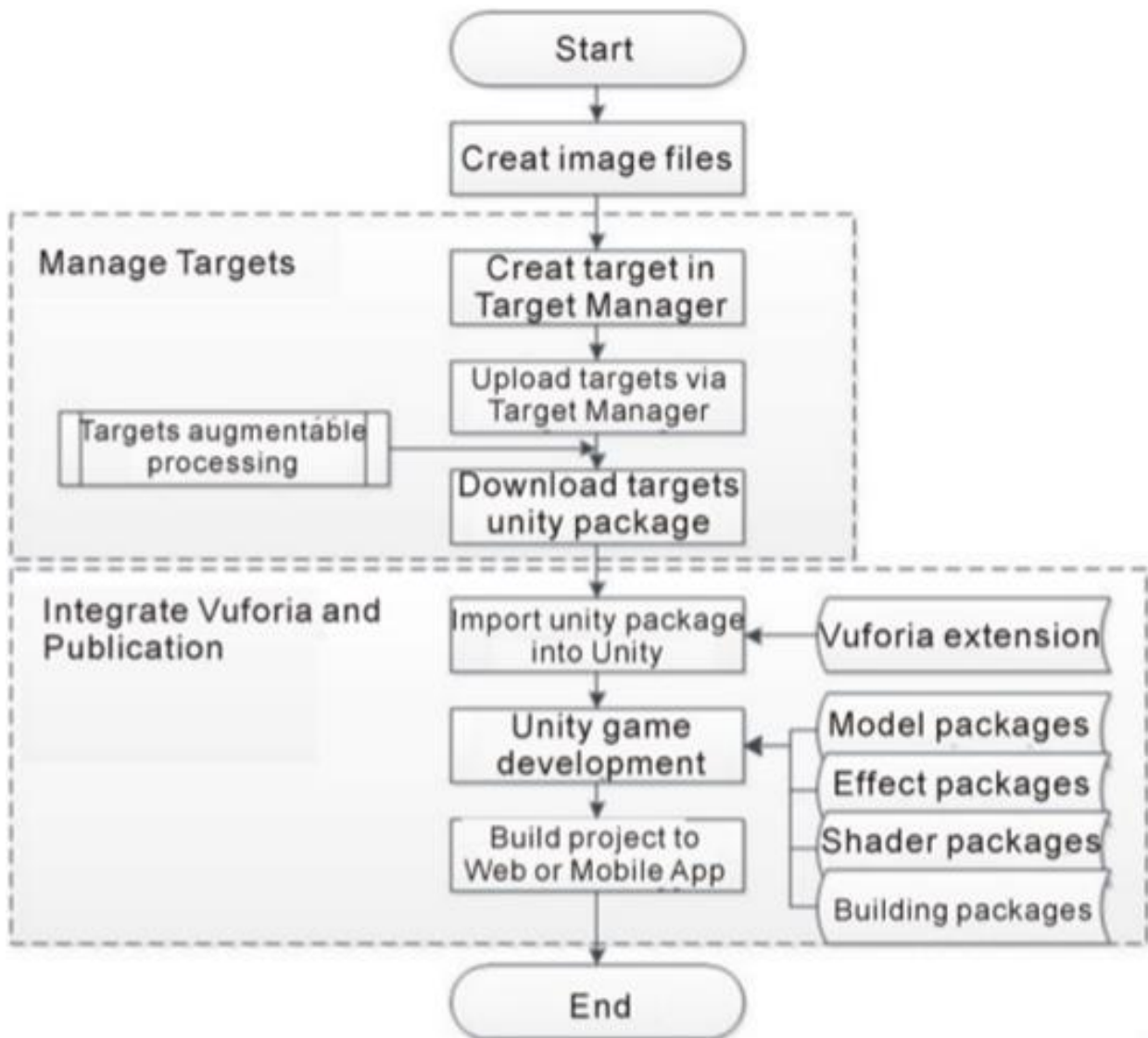
مراحل عملکرد Vuforia به شکل زیر می باشد:

Vuforia SDK



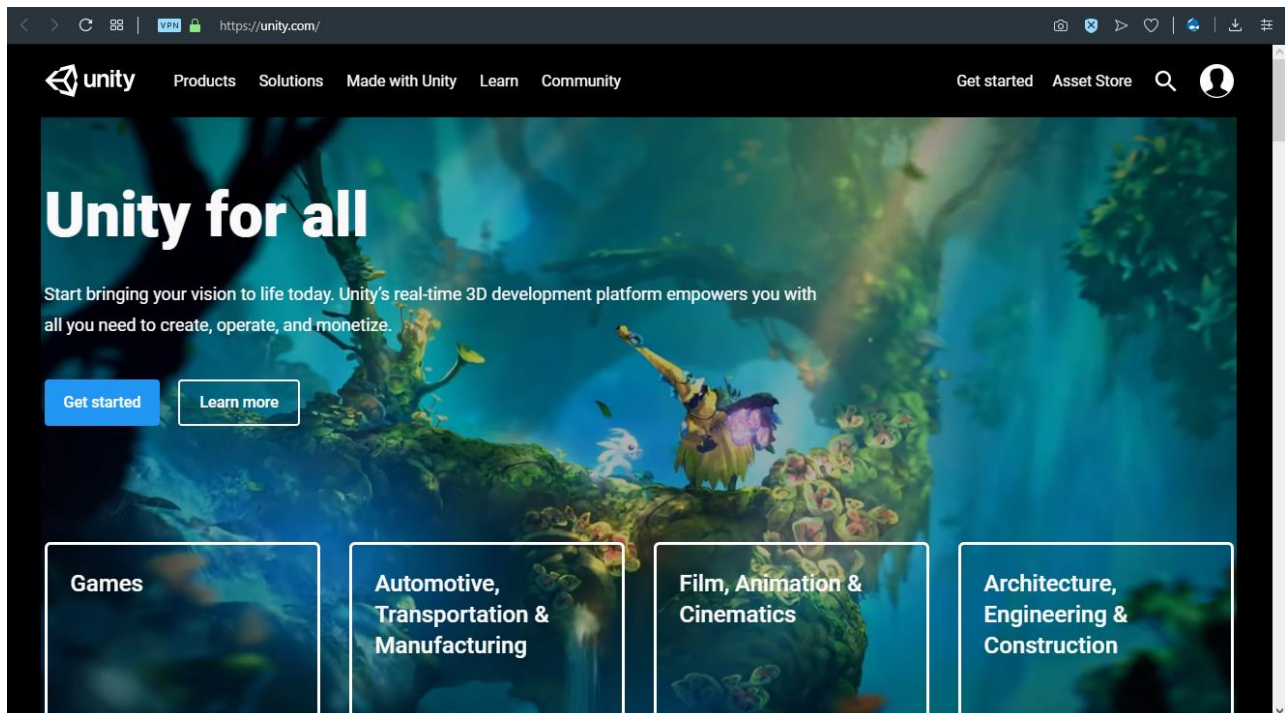
واقعیت افزوده Vuforia تصاویر را از دوربین دستگاه دریافت و آن ها را پیکسل به پیکسل تجزیه می کند. پیکسل ها برای پیدا کردن شباهت بین نقاط تعریف شده در تصاویر بانک و تصاویر دوربین بررسی شده و در صورت هم خوانی با هدف های تعریف شده کاربر نتیجه مثبت را بر می گردانند. اهداف می توانند انواع مختلف دو بعدی و سه بعدی و حتی اسکن اشیاء واقعی به صورت ترکیبی در بانک قرار داشته باشند. فریم ورک Vuforia نیز قابلیت های آماده ای مانند تعامل با واقعیت افزوده و قابلیت پشتیبانی از AR Core و AR Kit تعبیه شده را دارا می باشد. از فواید استفاده از AR Core بهبود تشخیص و عملکرد بهتر دوربین است زیرا برای دستگاه تفکیک سطوح صاف مانند زمین و تشخیص اشیاء مرتبط به آن مانند درخت، تعریف شده می باشد. خارج از زمینه تشخیص تصاویر، تمام کار های پردازش گرافیک و نمایش واسط کاربری و در کل تولید برنامه نهایی به عهده Unity3D می باشد. این فریم ورک در تلاش است تا با افزایش کارایی به صورت روزمره و بروز رسانی های مستمر، پیاده سازی منطق های برنامه نویسی در محیط واقعیت افزوده را ساده کند و به قدری در آن موفق بوده که به یک شیوه رایج تولید برنامه واقعیت افزوده تبدیل شده است.

فرآیند تولید یک برنامه واقعیت افزوده با استفاده از Vuforia در Unity3D را می توان در نمودار زیر خلاصه کرد:

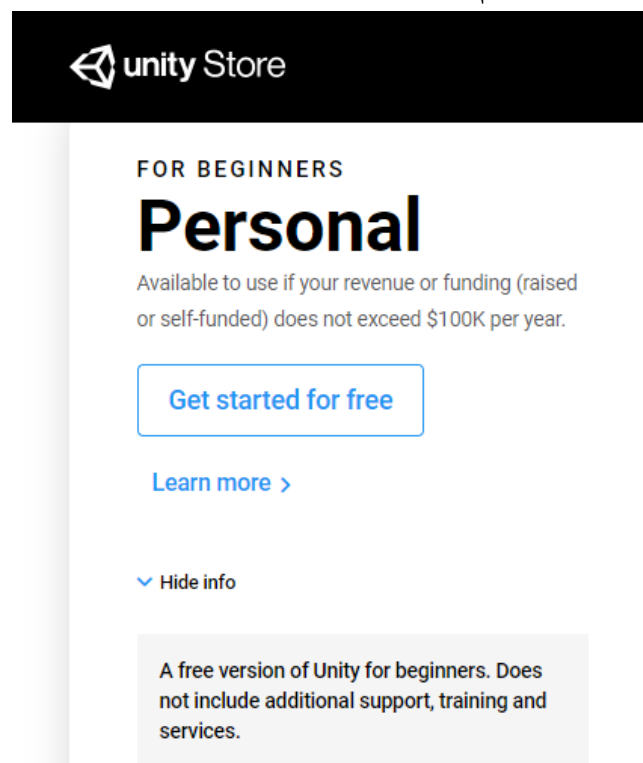


نصب و راه اندازی

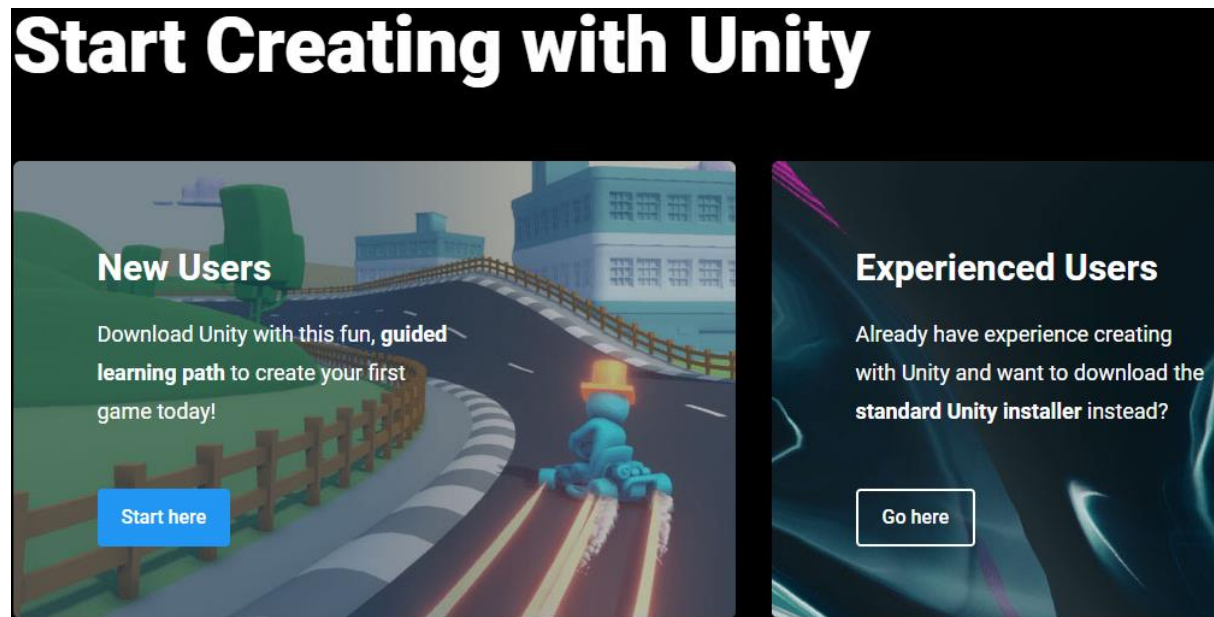
برای باز کردن و تغییر کد منبع پروژه نیاز به دریافت نرم افزار Unity3D به همراه افزونه Vuforia است. **نسخه Unity3D به کار رفته همراه دیسک پروژه ارائه شده و نیاز به دانلود مجدد نیست؛** صرفاً جهت آموزش نحوه دانلود و نصب توضیح داده می شود. ابتدا به سایت <https://unity.com> می رویم:



بر روی گزینه Get started کلیک می کنیم. در ادامه گزینه Get started for free را انتخاب می کنیم:



دکمه Start here را انتخاب می کنیم:



سپس Agree and download را می زنیم:

Terms

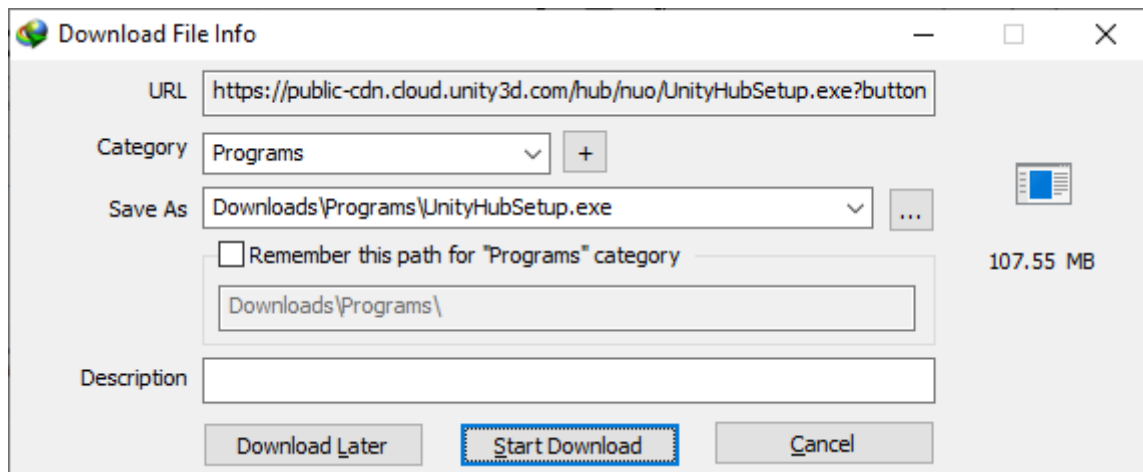


By downloading Unity, I confirm that I am eligible to use Unity Personal per the Terms of Service, as I or my company meet the following criteria:

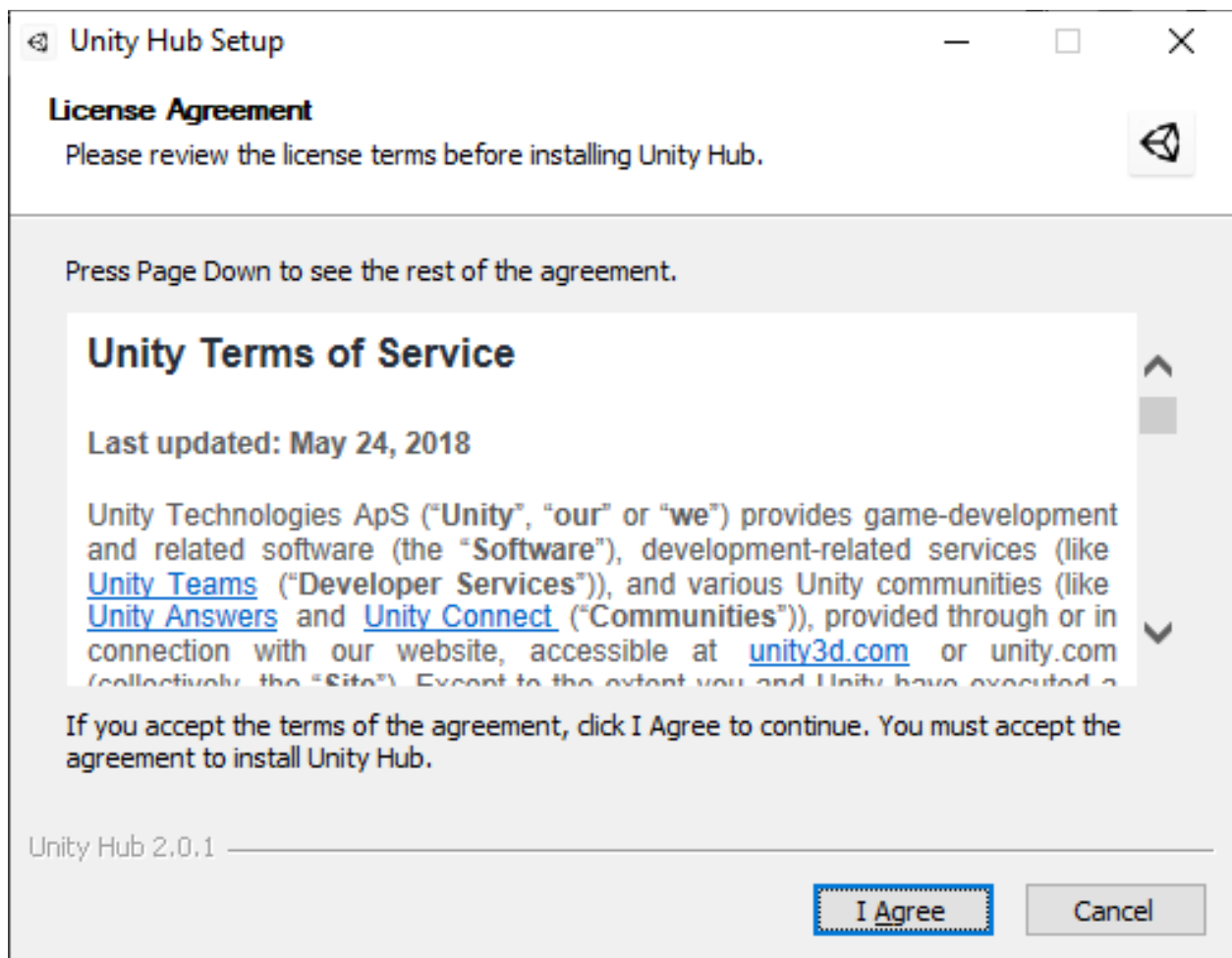
- Do not make more than \$100k in annual gross revenues, regardless of whether Unity Personal is being used for commercial purposes, or for an internal project or prototyping.
- Have not raised funds in excess of \$100K.
- Not currently using Unity Plus or Pro.
- Have read and acknowledged Unity's [Privacy Policy](#)

Agree and download

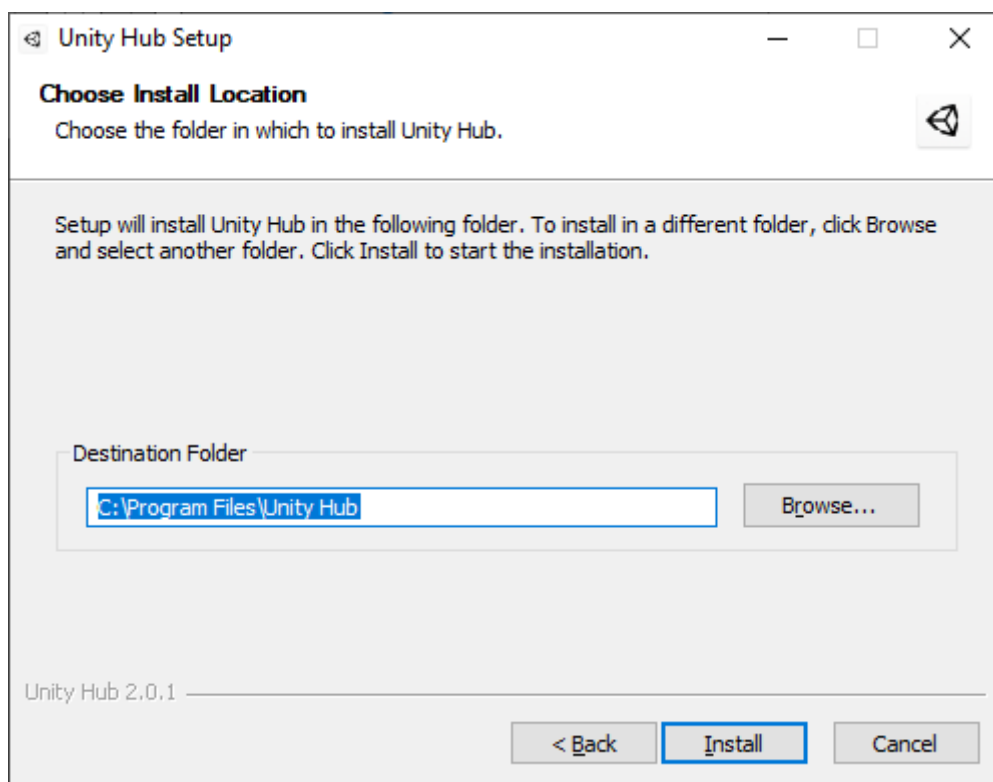
نرم افزار Unity Hub دانلود می شود:



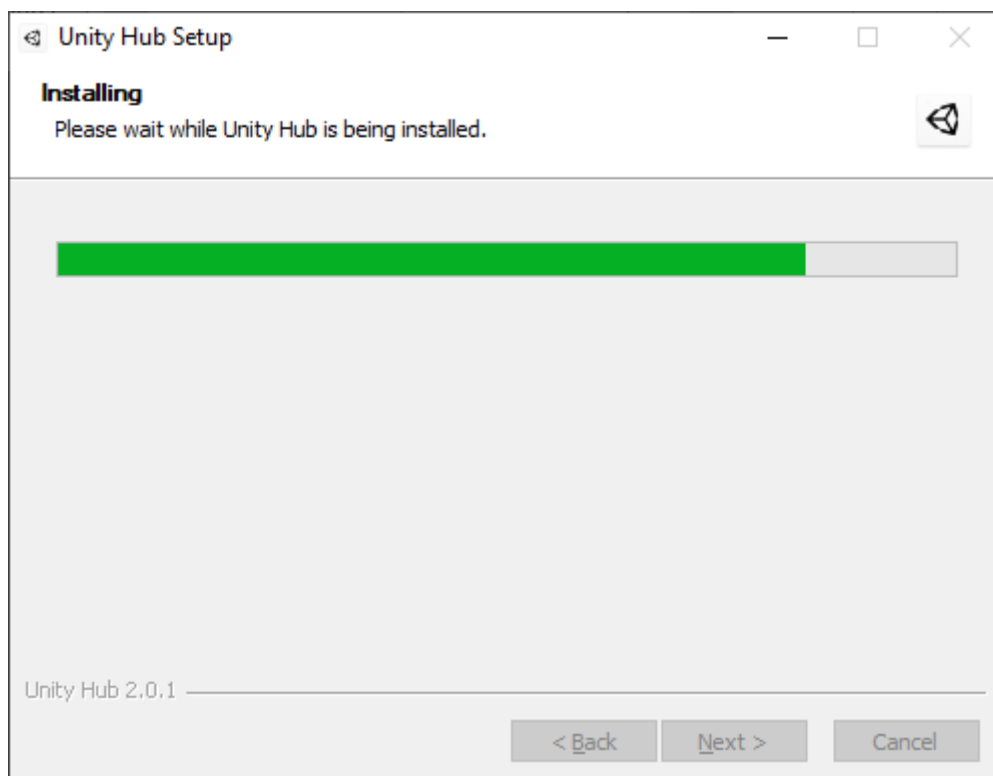
مراحل نصب را طی می کنیم:



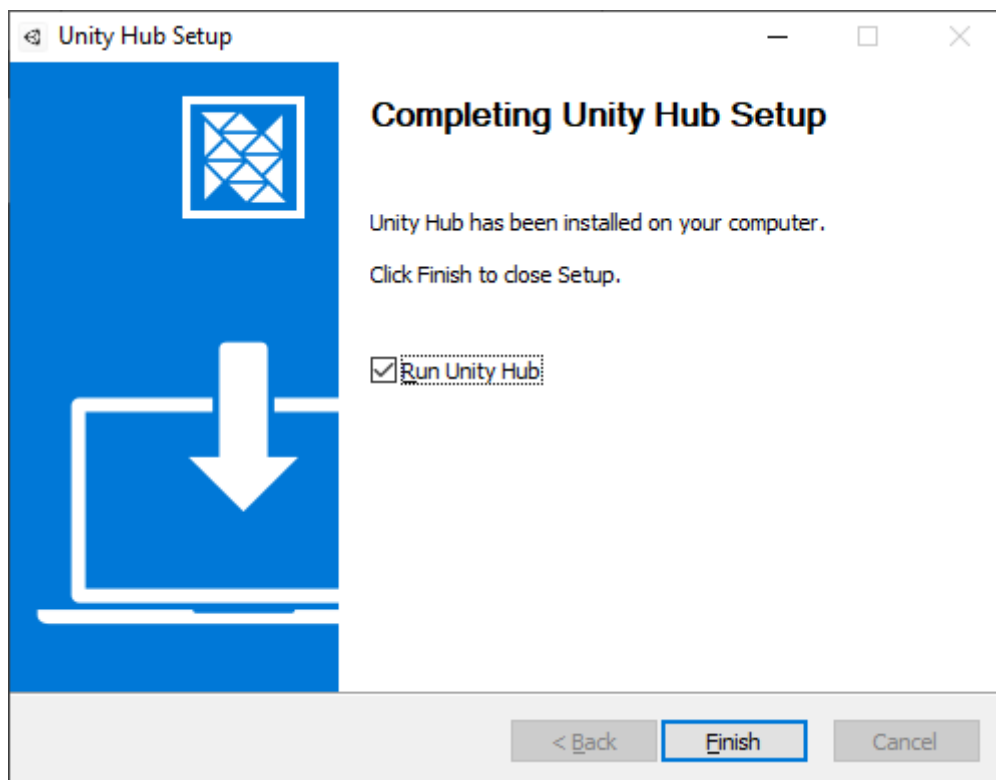
گزینه Install:



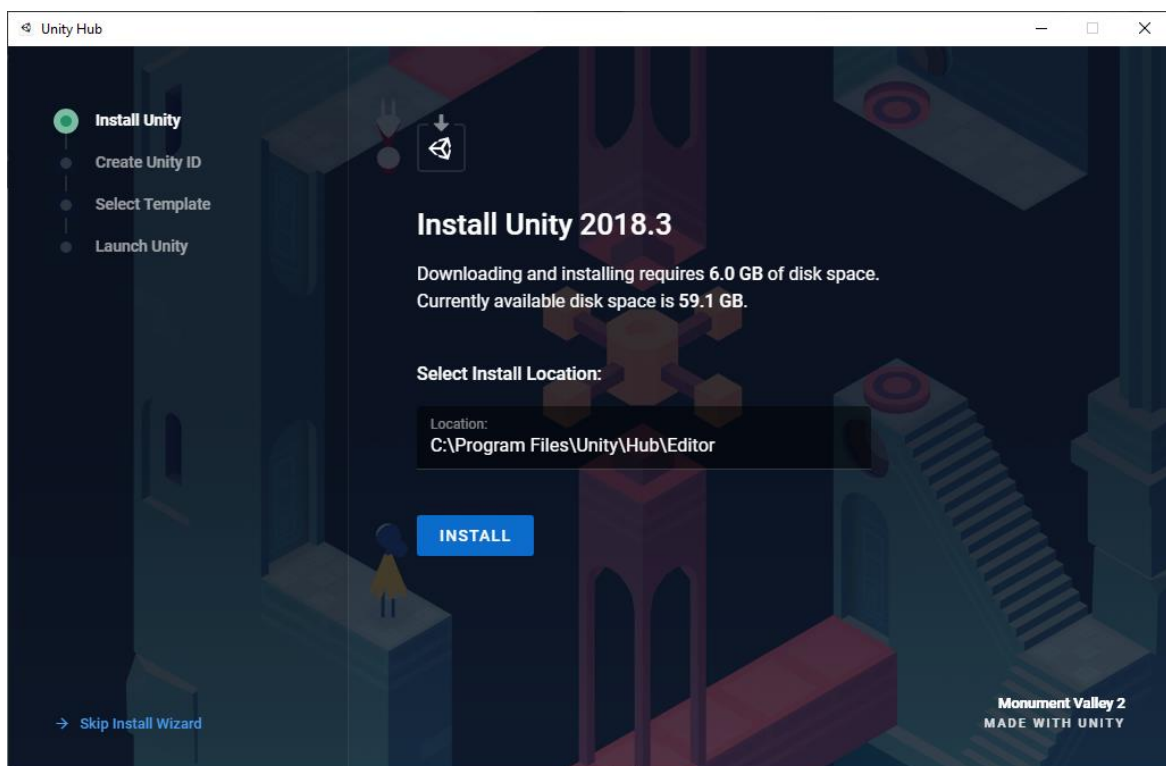
تا پایان نصب منتظر می مانیم:



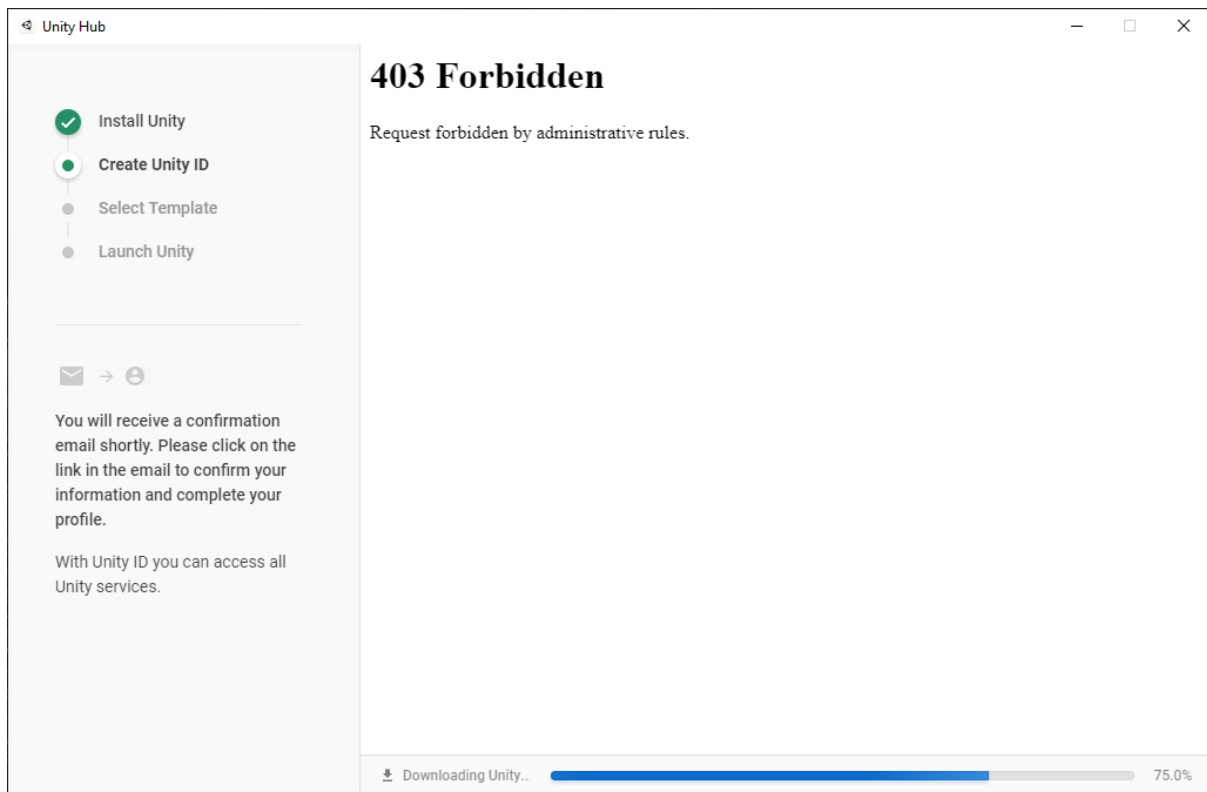
حال برنامه آماده اجرا است:



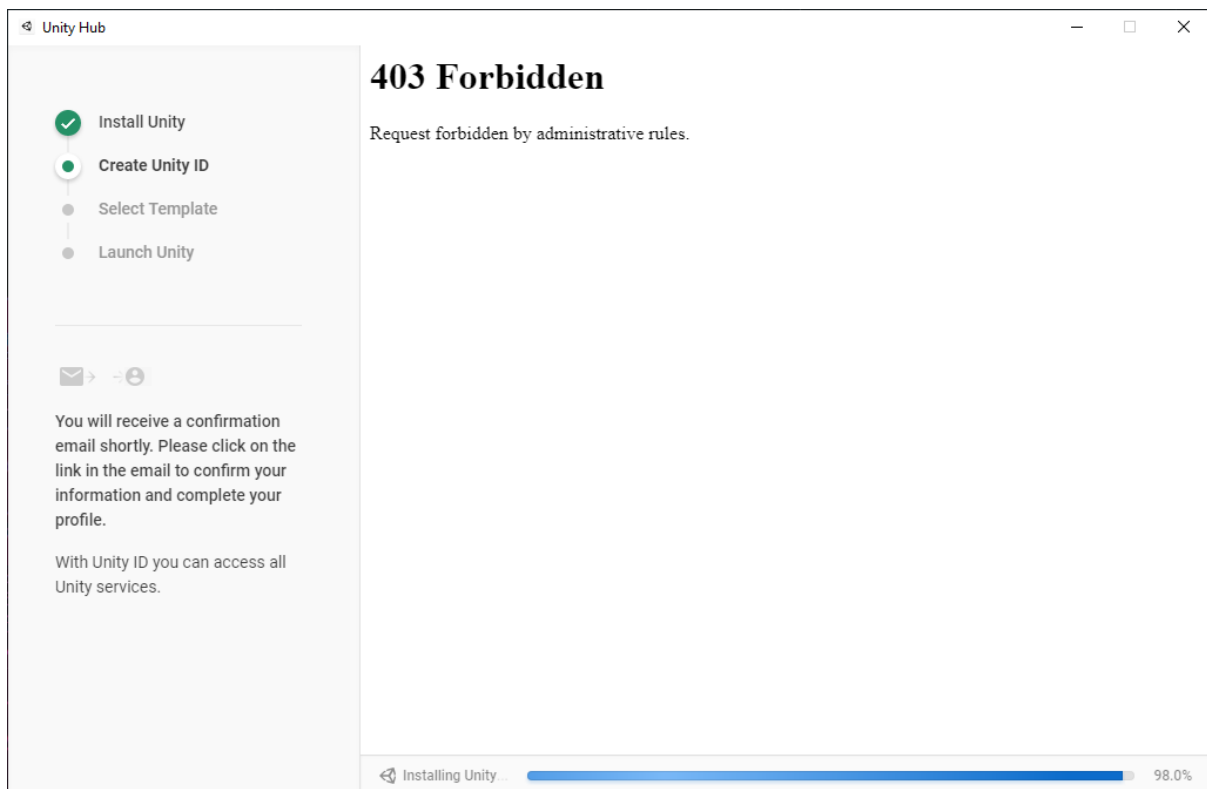
برای استفاده از نسخه داخل دیسک گزینه **Skip install wizard** را از پایین سمت چپ انتخاب کنید. دانلود موتور Unity3D را آغاز می کنیم:



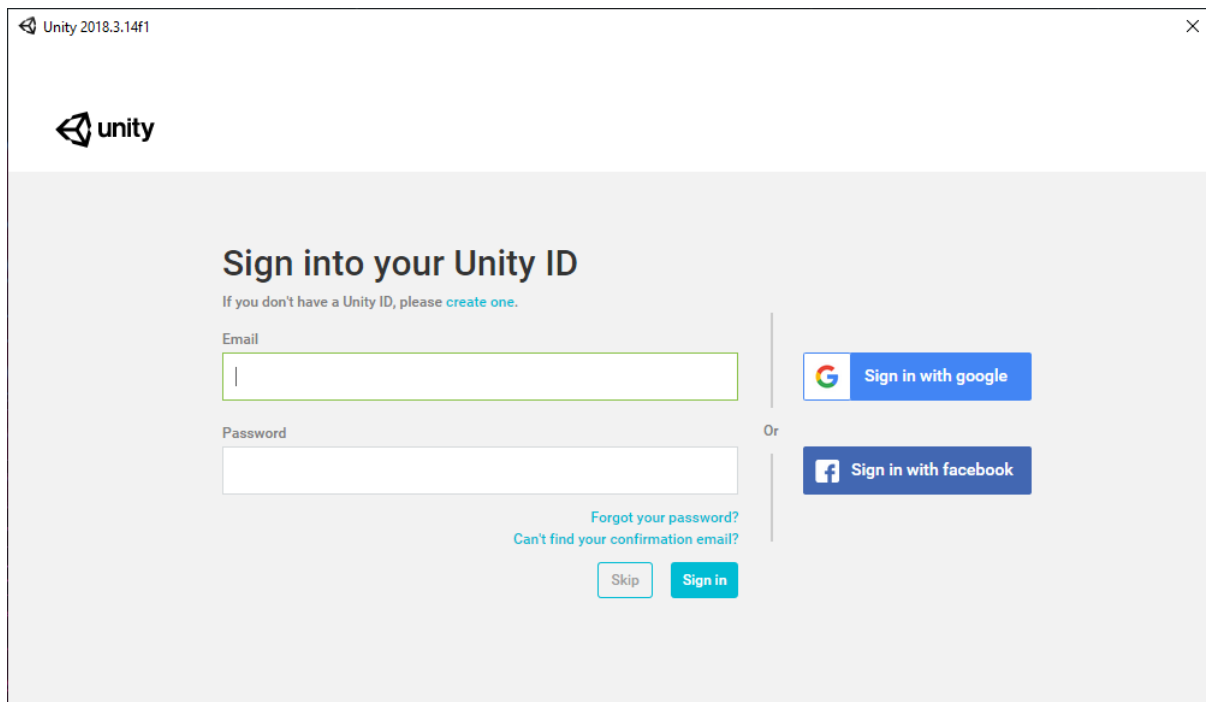
و تا اتمام دانلود منتظر می مانیم:



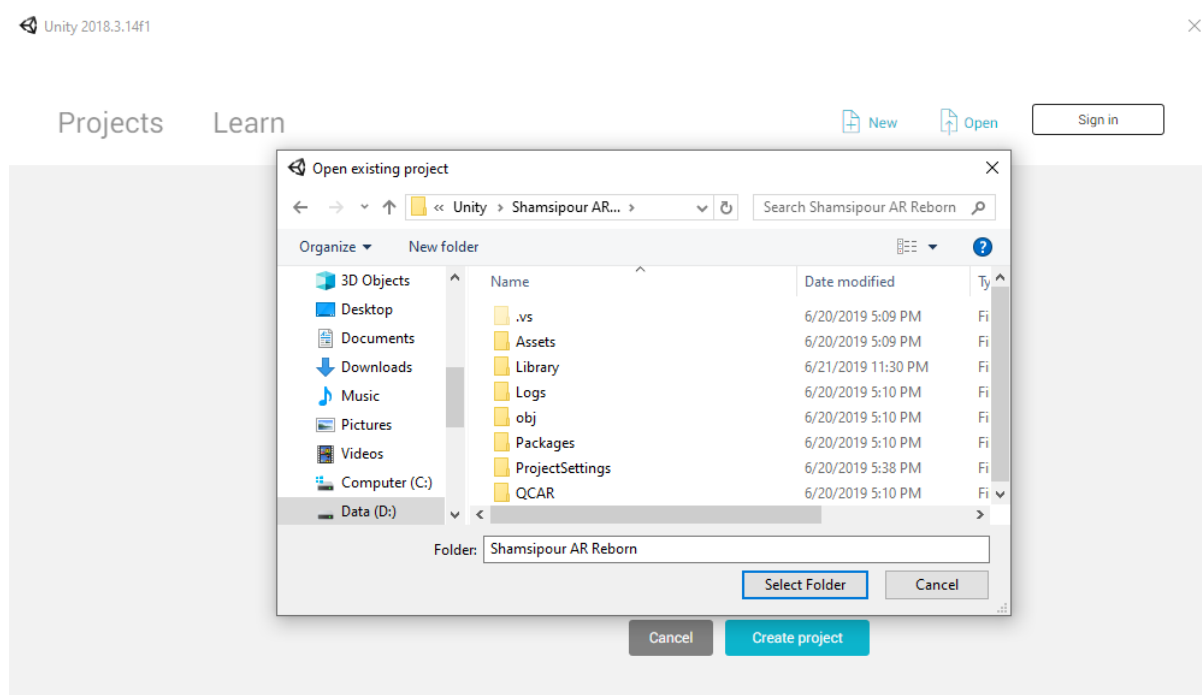
همزمان با دانلود، عملیات نصب صورت می گیرد:



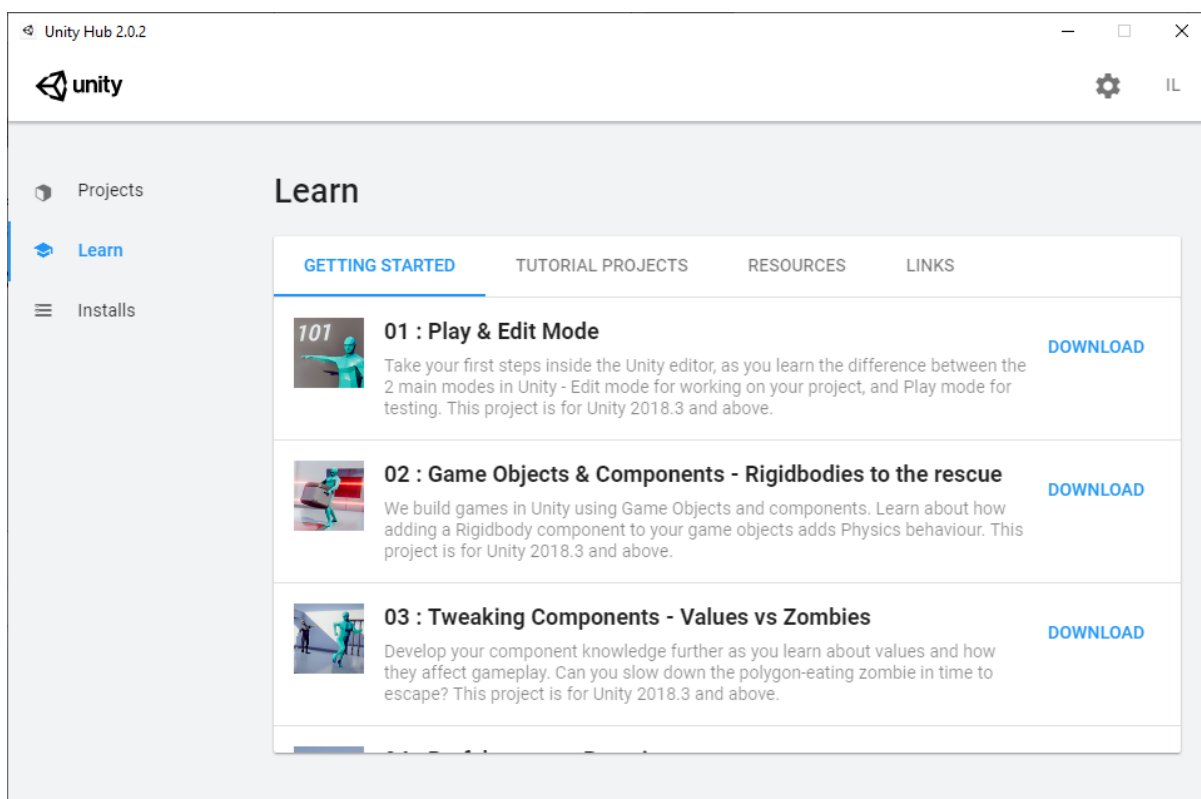
نرم افزار را اجرا کرده و با حساب کاربری گوگل یا Unity وارد شده و یا می توانیم این مرحله را Skip کنیم:



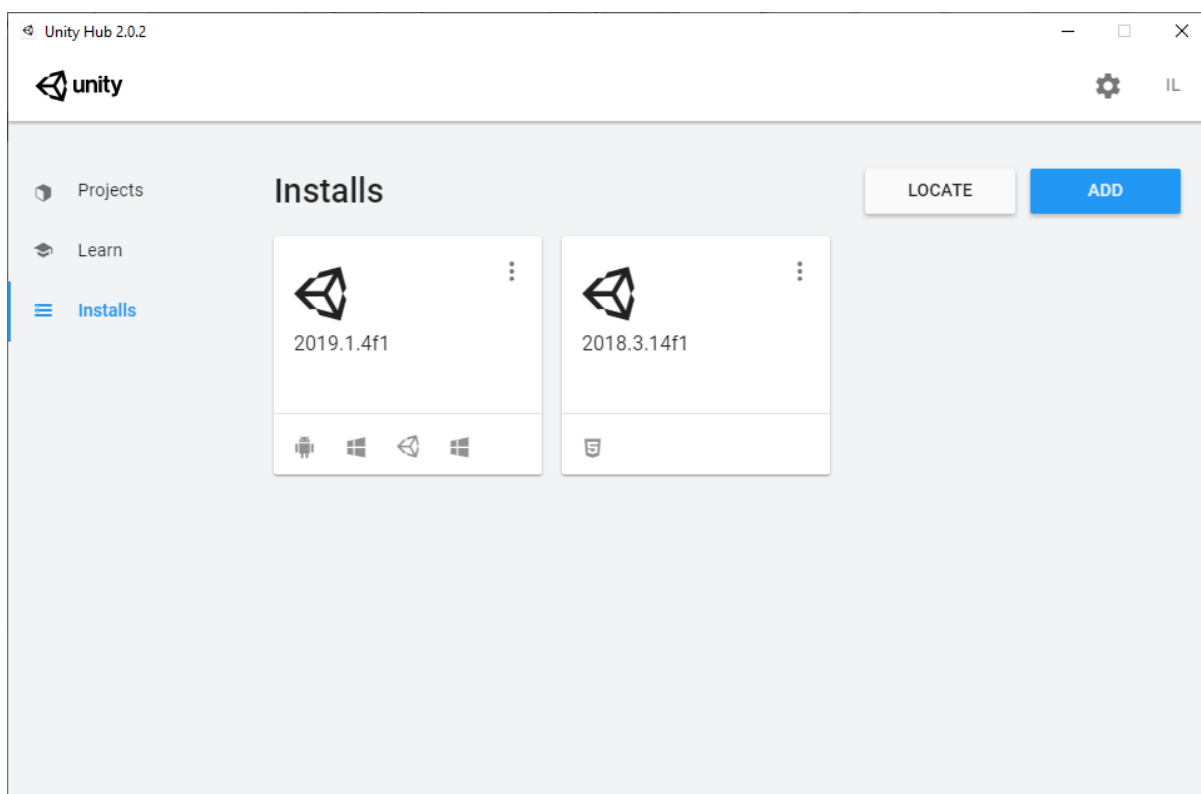
از گوشه سمت راست بالا گزینه Open و پوشه پروژه را انتخاب می کنیم:



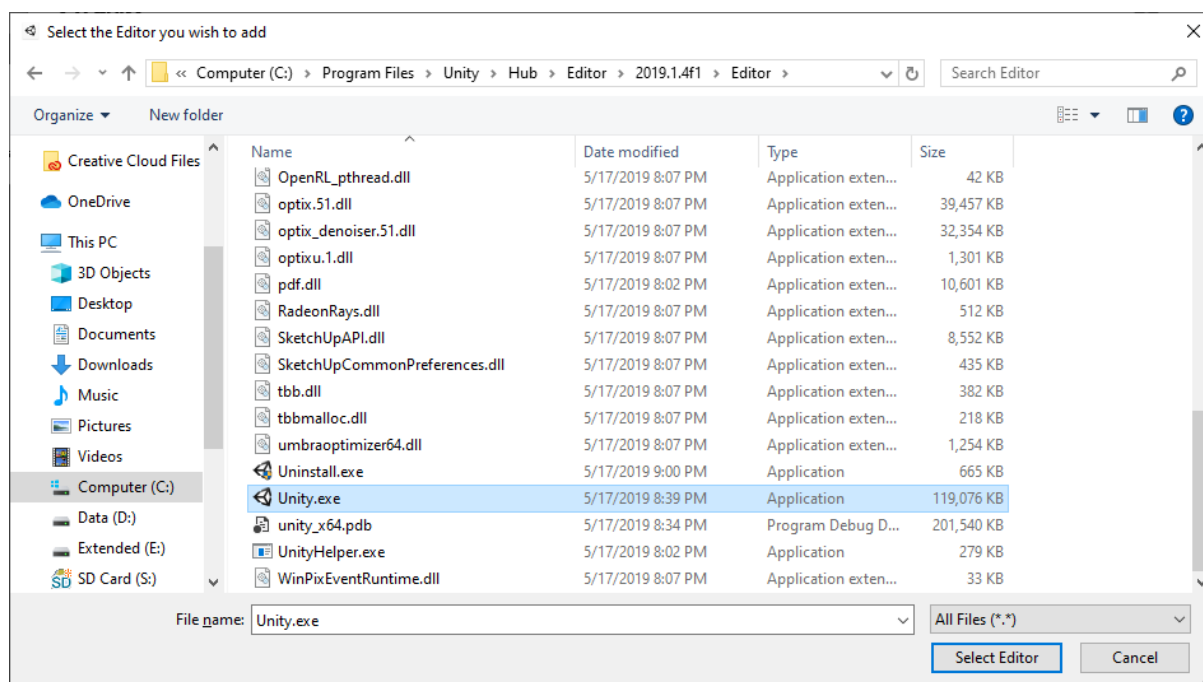
برای استفاده از نسخه داخل دیسک، Unity Hub را اجرا و Skip install wizard را انتخاب می کنیم و با پنجره زیر مواجه می شویم:



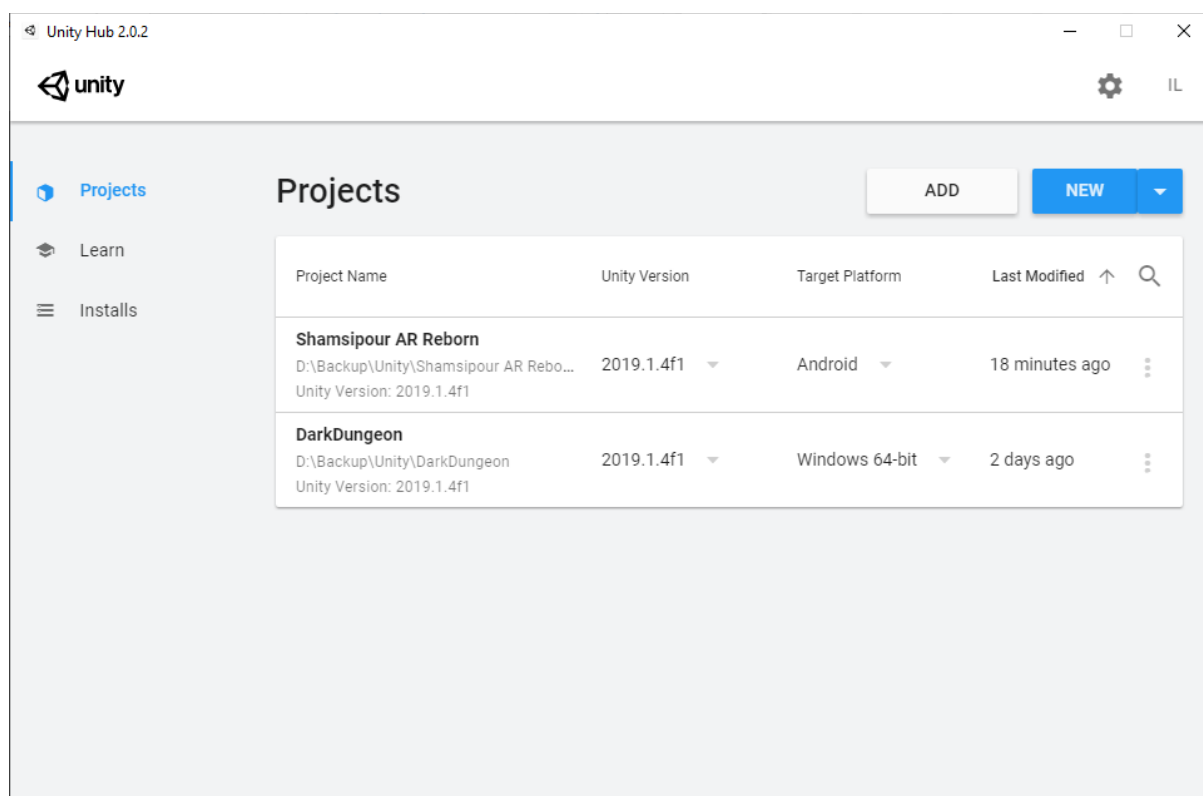
در منوی سمت چپ به بخش Install رفته و از بالا سمت راست گزینه Locate را انتخاب می کنیم:



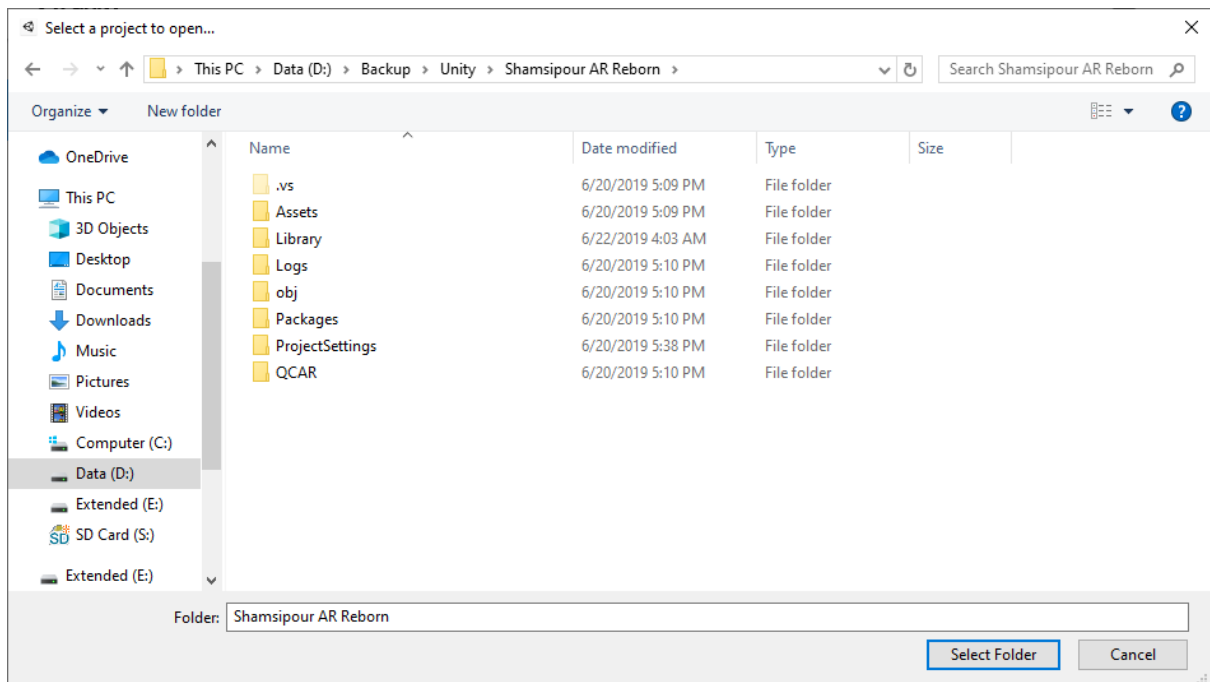
برای اضافه کردن نسخه Unity3D مورد نیاز پروژه به Unity Hub، کافی است فایل اجرایی Unity را از داخل دیسک یا در محل کپی شده انتخاب کنید تا از این پس پروژه های Unity با این نسخه اجرا شوند:



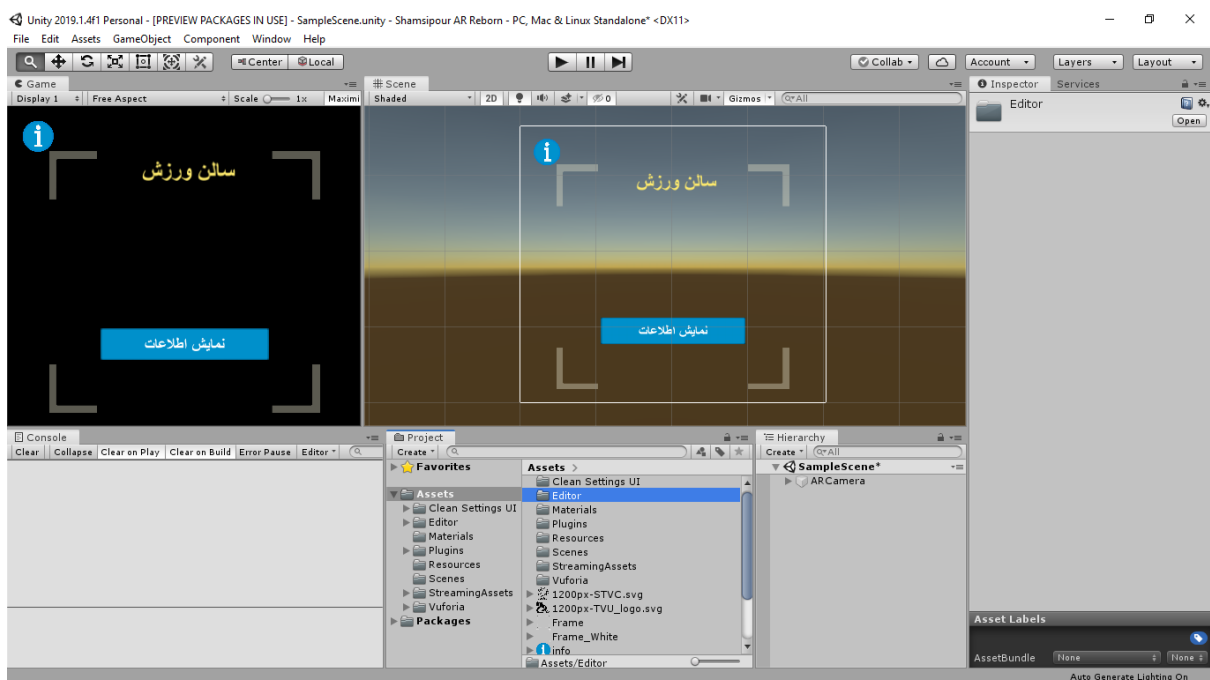
نسخه Unity در بخش Installs برنامه Unity Hub ظاهر می شود. حال برای باز کردن پروژه از همان منوی سمت چپ به بخش Projects می رویم:



در سمت راست بالا گزینه Add را انتخاب و آدرس پروژه را می دهیم:



حال باید مانند تصویر قبل، نام پروژه در لیست ظاهر شود. با دو بار کلیک بر روی نام آن پروژه اجرا خواهد شد:



برای گرفتن خروجی متناسب با دستگاه مورد نظر، نیاز به دانلود SDK آن پلتفرم و تغییر آدرس های برنامه از منوی بالا Edit زیر منوی Preferences و در پنجره باز شده بخش External Tools پایین صفحه است. برای مثال ابزار توسعه اندروید همراه Android Studio نصب می شوند.

مشخصات سیستم مورد نیاز

برای توسعه در Unity:

- سیستم عامل ویندوز 7، 8، 10 (فقط نسخه های 64 بیت)، مک 10.12 و جدیدتر
- پردازنده مرکزی با پشتیبانی از مجموعه دستورات SSE2
- گرافیک سازگار با DirectX 11 و پشتیبانی از Shader Model 4.0

برای توسعه پلتفرم:

- خروجی iOS نیاز به macOS نسخه 10.12.6 و Xcode نسخه 9.4 و به بالا دارد
- خروجی Android نیاز به SDK اندروید و JDK دارد. دریافت NDK نیز توصیه می شود.
- خروجی UWP فقط بر روی Windows 10 و فقط نسخه های 64-bit ممکن است. برای کامپایل کردن نصب Visual Studio نسخه 2015 و یا جدیدتر به همراه ابزار توسعه C++ و همچنین Windows 10 SDK نیاز است.

برای اجرای پلتفرم:

- دستگاه های اپل باید به iOS نسخه 9 و بالا تر مجهز باشند.
- دستگاه های اندروید باید نسخه 4.1 به بالا و دارای پردازنده Atom و یا ARMv7 با قابلیت پشتیبانی NEON باشند. برای پردازش گرافیکی نیز به OpenGL ES 2.0 نیاز است.
- دستگاه های قادر به اجرای UWP باید به Windows 10 و DirectX 10 مجهز باشند.

بانکها و جداول

درون بانک برای هر تصویر یک جدول از نوع QCAR که ساختار Code First دارد قرار داده شده است و به شرح زیر می باشد:

بانک QCAR:

```
<QCARConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="qcar_config.xsd">
```

جداول:

<Tracking>

```
<ImageTarget name="Farhang" size="2.500000 0.742718" />
<ImageTarget name="Asli" size="4.500000 1.426246" />
<ImageTarget name="Entesharat" size="1.000000 0.814770" />
<ImageTarget name="Fani" size="1.300000 0.772906" />
<ImageTarget name="Tarbiat" size="2.000000 0.919301" />
<ImageTarget name="Self" size="2.000000 0.657301" />
<ImageTarget name="Boofe" size="2.000000 1.071630" />
<ImageTarget name="Sanat" size="2.000000 0.827314" />
<ImageTarget name="Andishe" size="2.000000 0.628407" />
<ImageTarget name="Ghadim" size="4.000000 3.469301" />
<ImageTarget name="MIS" size="1.500000 1.007812" />
```

</Tracking>

مقصود این اعداد کمک به دوربین برای پیدا کردن اندازه (متر) دقیق تصویر پردازش شده برای هر موجودیت می باشد که به اصطلاح به آن Tracking می گویند.

فرا داده بانک:

fileFormatVersion: 2

guid: 9814c65b09764294ab9cd3131f6c5725

folderAsset: yes

DefaultImporter:

externalObjects: {}

هر تصویر شامل فرا داده خود می شود، هر فرا داده به موتور Unity3D برای درک بهتر مشخصات تصاویر کمک می کند. مشخصات اولیه شناسایی تصویر:

fileFormatVersion: 2

guid: 4d4ff3f4cdb0494895568ff5bd432183

TextureImporter:

fileIDToRecycleName{}:

externalObjects{}:

serializedVersion: 9

مشخصات Mapping^{۱۲} تصویر:

mipmaps:

mipMapMode: 0

enableMipMap: 1

sRGBTexture: 1

linearTexture: 0

fadeOut: 0

borderMipMap: 0

mipMapsPreserveCoverage: 0

alphaTestReferenceValue: 0.5

mipMapFadeDistanceStart: 1

mipMapFadeDistanceEnd: 3

bumpmap:

convertToNormalMap: 0

externalNormalMap: 0

heightScale: 0.25

normalMapFilter: 0

isReadable: 0

streamingMipmaps: 0

streamingMipmapsPriority: 0

¹² Method for defining high frequency detail, surface texture, or color information

مشخصات رنگ و اندازه در Mapping:

```
grayScaleToAlpha: 0
generateCubemap: 6
cubemapConvolution: 0
seamlessCubemap: 0
textureFormat: 1
maxTextureSize: 2048
textureSettings:
  serializedVersion: 2
  filterMode: -1
  aniso: -1
  mipBias: -100
  wrapU: -1
  wrapV: -1
  wrapW: -1
  nPOTScale: 1
```

مشخصات نورپردازی تصاویر:

```
lightmap: 0
compressionQuality: 50
spriteMode: 0
spriteExtrude: 1
spriteMeshType: 1
```

مشخصات قرارگیری نور و رنگ در تصاویر:

```
alignment: 0
spritePivot: {x: 0.5, y: 0.5}
spritePixelsToUnits: 100
spriteBorder: {x: 0, y: 0, z: 0, w: 0}
spriteGenerateFallbackPhysicsShape: 1
```


مشخصات کانال تصویر:

alphaUsage: 1
alphaIsTransparency: 0
spriteTessellationDetail: -1
textureType: 0
textureShape: 1
singleChannelComponent: 0
maxTextureSizeSet: 0
compressionQualitySet: 0
textureFormatSet: 0

مشخصات تصویر در محیط موتور:

platformSettings:
- serializedVersion: 2
buildTarget: DefaultTexturePlatform
maxTextureSize: 2048
resizeAlgorithm: 0
textureFormat: -1
textureCompression: 0
compressionQuality: 50
crunchedCompression: 0
allowsAlphaSplitting: 0
overridden: 0
androidETC2FallbackOverride: 0

spriteSheet:

serializedVersion: 2

sprites[]:

outline[]:

physicsShape[]:

bones[]:

spriteID:

vertices[]:

indices:

edges[]:

weights[]:

spritePackingTag:

pSDRemoveMatte: 0

pSDShowRemoveMatteOption: 0

ساختمان داده:

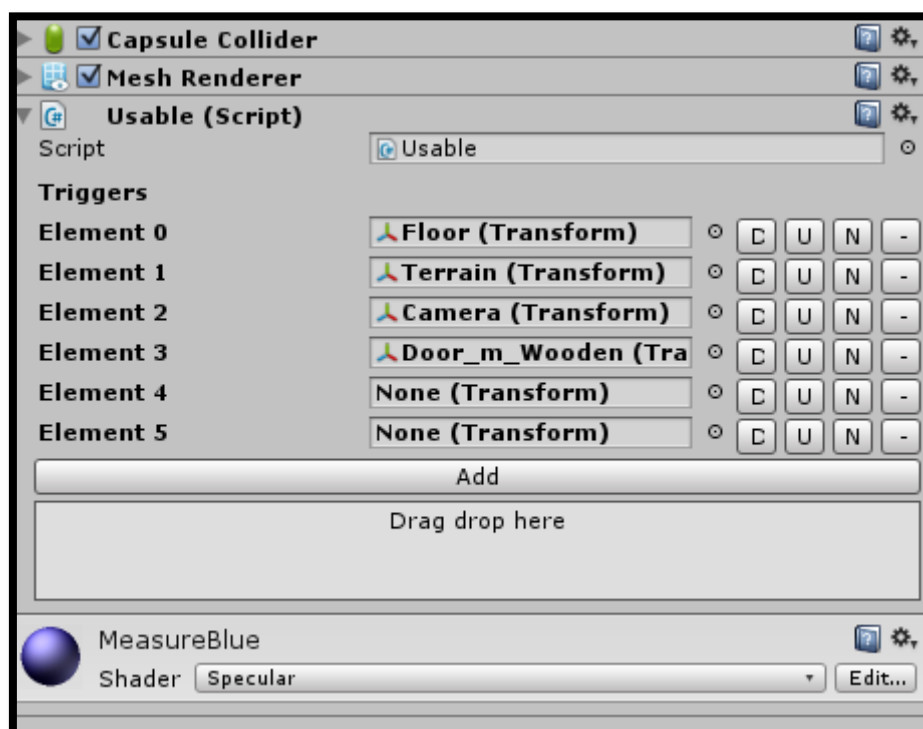
ساختمان داده برنامه از نوع ArrayList که در C# خود زیر مجموعه List های Generic یا List<T> است می باشد. تفاوت بارز ArrayList این است که محدود به یک نوع خاص از داده نیست و چندین نوع را می تواند در خود نگه دارد پس برای ذخیره کردن Object ها مناسب است.

```
using UnityEngine;
using System.Collections;

public class Arrays : MonoBehaviour
{
    public ArrayList inventory = new ArrayList();

    void Start()
    {
        inventory.Add(100);
        inventory.Add("Name");
        inventory.Add(GameObject.Find("ObjName"));
    }
}
```

داخل محیط موتور Unity3D هر خانه ساختمان داده به صورت یک المان قابل تغییر نمایش داده می شود:



دستورات برنامه

دستورات ابتدایی برنامه و تعاریف اعضای حفاظت شده:

```
using UnityEngine;
using Vuforia;
using UnityEngine.UI;

public class DefaultTrackableEventHandler : MonoBehaviour, ITrackableEventHandler
{
    private GameObject scan_Color;
    void Awake()
    {
        scan_Color = GameObject.Find("Scanner");
    }

    protected TrackableBehaviour mTrackableBehaviour;
    protected TrackableBehaviour.Status m_PreviousStatus;
    protected TrackableBehaviour.Status m_NewStatus;

    protected virtual void Start()
    {
        mTrackableBehaviour = GetComponent<TrackableBehaviour>();
        if (mTrackableBehaviour)
            mTrackableBehaviour.RegisterTrackableEventHandler(this);
    }
}
```

رابط MonoBehaviour برای استفاده از توابع داخلی کتابخانه UnityEngine و ITrackableEventHandler رابط تعریف شده فریم ورک Vuforia می باشد. scan_Color جهت تعویض رنگ فریم در برنامه می باشد و توسط تابع Awake شی با Tag نام Scanner در موتور پیدا شده (شی همان فریم) و آن را به scan_Color تخصیص می دهد. کلاس TrackableBehaviour یک کلاس انتزاعی در کتابخانه Vuforia و از نوع Serialize شده است. تابع Start یک شی تعریف شده mTrackableBehaviour را از طریق GetComponent وضعیت آن را دریافت و در صورت کامل دریافت شدن، از RegisterTrackableEventHandler که یک تابع دیگر کتابخانه Vuforia با یک آرگومان ITrackableEventHandler می باشد آن را به اصطلاح باندل می کند. خلاف همین عمل در تابع OnDestroy صورت می پذیرد.

```
protected virtual void OnDestroy()
{
    if (mTrackableBehaviour)
        mTrackableBehaviour.UnregisterTrackableEventHandler(this);
}
```

در ادامه وضعیت قرار گیری تصویر در فریم بررسی می شود

```
public void OnTrackableStateChanged(
    TrackableBehaviour.Status previousStatus,
    TrackableBehaviour.Status newStatus)
{
    m_PreviousStatus = previousStatus;
    m_NewStatus = newStatus;

    Debug.Log("Trackable " + mTrackableBehaviour.TrackableName +
        " " + mTrackableBehaviour.CurrentStatus);

    if (newStatus == TrackableBehaviour.Status.DETECTED ||
        newStatus == TrackableBehaviour.Status.TRACKED ||
        newStatus == TrackableBehaviour.Status.EXTENDED_TRACKED)
    {
        scan_Color.GetComponent<UnityEngine.UI.RawImage>().color = new Color(0, 245, 0, 190);
        OnTrackingFound();
    }
    else if (previousStatus == TrackableBehaviour.Status.TRACKED &&
        newStatus == TrackableBehaviour.Status.NO_POSE)
    {
        scan_Color.GetComponent<UnityEngine.UI.RawImage>().color = new Color(255, 237, 109, 214);
        OnTrackingLost();
    }
    else
    {
        OnTrackingLost();
    }
}
```

سه حالت در شناسایی تصویر بررسی می شود. حالت DETECTED و TRACKED برای قرار گیری تصویر در فریم به صورت عادی و EXTENDED_TRACKED ادامه فعال بودن حالت اول حتی پس از گذشت مدتی کوتاه پس از توقف همخوانی تصویر پردازش شده می باشد. TRACKED به معنای فعال بودن دوربین و دریافت فریم به فریم تصاویر برای پردازش است و DETECTED به معنی همخوانی تصویر در بانک و تصویر دریافت شده می باشد. در صورت فعال شدن دو مورد اول، فریم تصویر به رنگ سبز درآمده و تابع OnTrackingFound فعال و در غیر این صورت اگر تنها حالت TRACKED و NO_POSE فعال باشد و یا هیچ کدام از حالت ها فعال نباشد، برنامه به کار خود ادامه خواهد داد. این حالات یک enum به شرح زیر می باشند:

```
public enum Status
{
    NO_POSE = 0,
    LIMITED = 1,
    DETECTED = 2,
    TRACKED = 3,
    EXTENDED_TRACKED = 4
}
```

تابع `OnTrackingFound` زمانی که تصویر پردازش شده شناسایی شود فعال شده و سه شیء از نوع کلاس `Renderer` و `Collider` و `Canvas` به همراه فرزندان آن دریافت می کند.

```
protected virtual void OnTrackingFound()
{
    if (mTrackableBehaviour)
    {
        var rendererComponents = mTrackableBehaviour.GetComponentsInChildren<Renderer>(true);
        var colliderComponents = mTrackableBehaviour.GetComponentsInChildren<Collider>(true);
        var canvasComponents = mTrackableBehaviour.GetComponentsInChildren<Canvas>(true);

        // Enable rendering:
        foreach (var component in rendererComponents)
            component.enabled = true;

        // Enable colliders:
        foreach (var component in colliderComponents)
            component.enabled = true;

        // Enable canvas':
        foreach (var component in canvasComponents)
            component.enabled = true;
    }
}
```

این کار جهت فعال شدن همه محتوای قرار گرفته در محیط موتور جهت افزوده شدن به محیط واقعی صورت می گیرد. در صورت پیدا نشدن تصویر، برنامه دستور غیر فعال کردن تمام فرزندان اشیاء `Canvas` و `Renderer` را ارسال می کند.

```
protected virtual void OnTrackingLost()
{
    if (mTrackableBehaviour)
    {
        var rendererComponents = mTrackableBehaviour.GetComponentsInChildren<Renderer>(true);
        var colliderComponents = mTrackableBehaviour.GetComponentsInChildren<Collider>(true);
        var canvasComponents = mTrackableBehaviour.GetComponentsInChildren<Canvas>(true);

        // Disable rendering:
        foreach (var component in rendererComponents)
            component.enabled = false;

        // Disable colliders:
        foreach (var component in colliderComponents)
            component.enabled = false;

        // Disable canvas':
        foreach (var component in canvasComponents)
            component.enabled = false;
    }
}
```

فعال شدن OnTrackingFound یک Scrollview را بصورت واقعیت افزوده ایجاد می کند. برای بستن آن در Scrollview_close.cs یک دستور به این صورت نوشته شده است:

```
void Update()
{
    if (Input.GetKeyDown(KeyCode.Escape))
    {
        GameObject.FindWithTag("Scroll_tag").SetActive(false);
    }
}
```

تابع Update در هر فریم برنامه یک بار فراخوانی می شود. برای خروج از برنامه، نیاز به فشردن دوبار متوالی دکمه بازگشت است:

```
public class Timing : MonoBehaviour
{
    float timeLeft = 10.0f;

    void Start()
    {
        Screen.fullScreen = !Screen.fullScreen;
    }

    void Update()
    {
        timeLeft -= Time.deltaTime;

        if (timeLeft <= 0)
        {
            StartCoroutine("Fade");
        }
    }

    void Fade()
    {
        GameObject.Find("Alert_Text").SetActive(false);
    }
}
```


و در پایان نیز برنامه به این صورت خارج می شود:

```
void Update()
{
    if(isQuiting == true)
    {
        if (Input.GetKeyDown (KeyCode.Escape) || Input.GetKeyDown(KeyCode.Space))
        {
            Debug.Log("Escape pressed time 02");
            Application.Quit();
        }
    }
    if (Input.GetKeyDown(KeyCode.Escape) || Input.GetKeyDown(KeyCode.Space))
    {
        Debug.Log("Escape pressed time 01");
        quitMessageGameObject.SetActive(true);
        isQuiting = true;
        StartCoroutine(QuitingTimer());
        //Application.Quit();
    }
}

IEnumerator QuitingTimer()
{
    yield return new WaitForSeconds(1);
    isQuiting = false;
    Debug.Log("No response");
    quitMessageGameObject.SetActive(false);
}
```

پیشنهاد

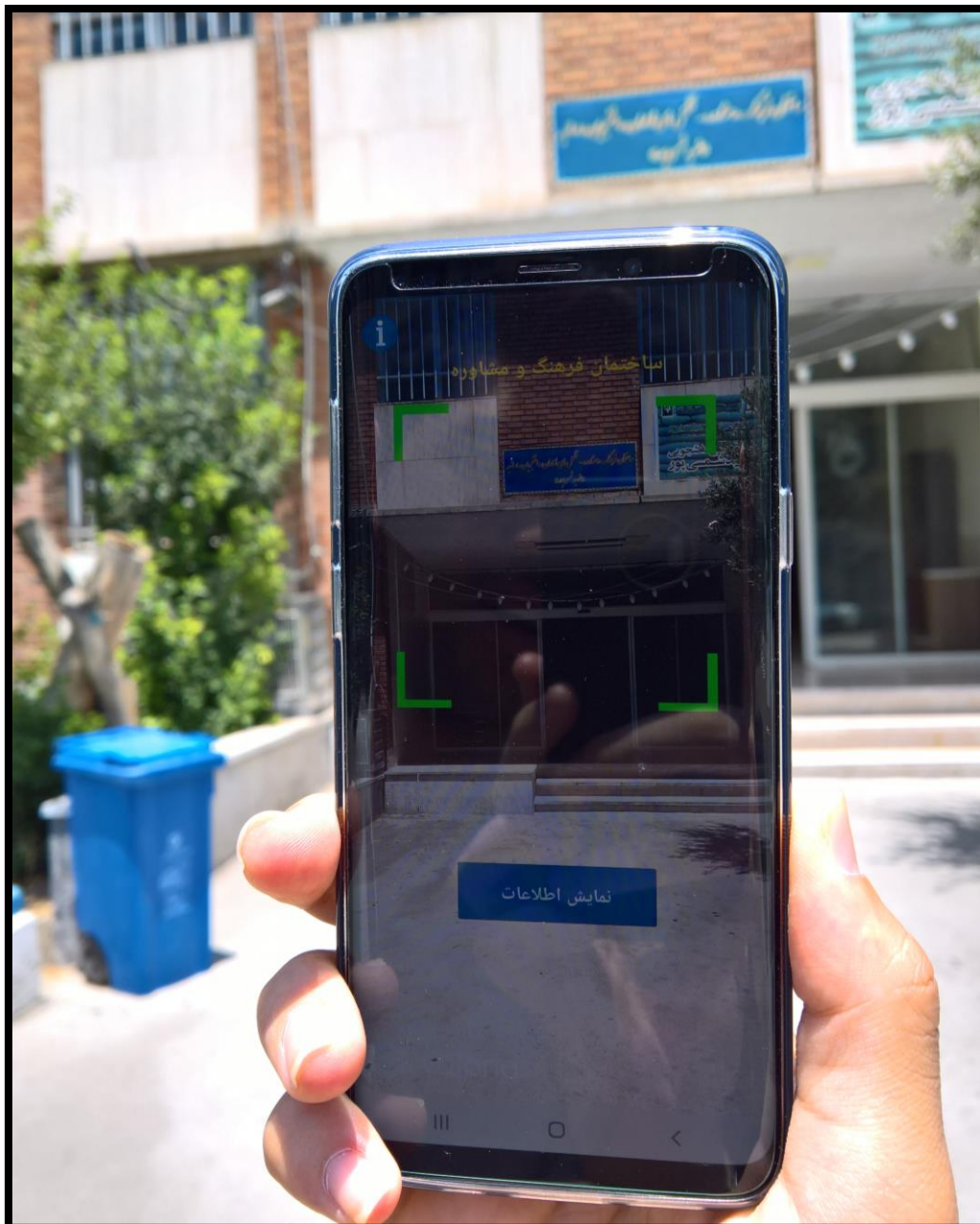
نرم افزار واقعیت افزوده دانشکده فنی و حرفه ای شهید شمس پور جهت فراگیری و بهره‌وری از فناوری واقعیت افزوده که صنعتی نوپا در علوم کامپیوتر است، توسعه داده شده و در نظر دارد تا دانشجویان با فراگیری این علم بتوانند از کد منبع نرم افزار که در اختیار عموم دانشجویان است جهت بهبود آن استفاده کنند. فراگیری علم و فن به کارگیری واقعیت افزوده شتیدا به علاقمندان توصیه می شود. به دانشجویان پیشنهاد می شود که بتوانند با نرم افزار فعلی، اماکن دیگر و اطلاعات تکمیلی بیشتر را به نرم افزار اضافه و تجربه نهایی واقعیت افزوده را بهتر کنند.

نتیجه گیری

استفاده از موتور گرافیکی بازی سازی معروف Unity3D در کنار فریم ورک قدرتمند واقعیت افزوده Vuforia AR می تواند محصولی فوق العاده را به ارمغان بیاورد. هدف اصلی واقعیت افزوده ارائه شیوه ای نوین برای تعامل کاربر با محیط پیرامون خود می باشد. نرم افزار هایی که به شیوه سنتی برای کار به خصوصی طراحی و توسعه می یابند، می توانند با ادغام قابلیت های واقعیت افزوده، تجربه ای سرگرم کننده و جذاب را به کاربر ارائه دهند. زمینه کاربرد واقعیت افزوده دامنه وسیعی از برنامه ها از جمله برنامه های مهندسی، علمی و آموزشی، سیستمی، هوش مصنوعی، اداری و ... را پوشش می دهد.

پیش بینی می شود تا سال ۲۰۲۵ میلادی^{۱۳}، ارزش سرمایه صنعت واقعیت افزوده به بیش از ۳۵ میلیارد دلار در زمینه های بازی های ویدیویی، سلامت، مهندسی، همایش های زنده، سرگرمی و املاک برسد و زمینه درآمد سازی قابل اطمینانی را فراهم کند. در حال حاضر در سال های ابتدایی پیشرفت واقعیت افزوده به سر می بریم و هنوز برای یادگیری و توسعه نرم افزار های درجه یک زمان زیادی باقی مانده است. امید بر آن است که این صنعت در سال های آتی در کشور عزیزمان ایران به طور قابل توجهی با همت اساتید، دانشجویان، مهندسان و تمامی علاقه‌مندان موجب پیشرفت و کارآفرینی در تمامی امور کشور شود.

¹³ <https://www.pcmag.com/news/363090/the-future-of-augmented-reality-is-serious-business>



نرم افزار را اجرا و در محیط دانشکده، دوربین را طوری در مقابل ساختمان ها قرار دهید که تابلوی آن ها مابین چهارچوب مشخص شده در برنامه قرار گیرند. سبز شدن چهارچوب به معنای شناخته شدن ساختمان می باشد. در صورت تمایل دکمه "نمایش اطلاعات" را لمس کنید تا اطلاعات و توضیحات ساختمان نمایش داده شود. با لمس صفحه و کشیدن انگشت به سمت بالا و یا پایین، متن را طومار کنید. در سمت چپ بالا، اطلاعات کلی نرم افزار قرار دارد. با یک بار لمس دکمه بازگشت به دوربین بازگشته و با دو بار لمس متوالی دکمه بازگشت از برنامه خارج شوید.

سایت توسعه دهنده:

<https://wscript.ir/>

دریافت کد منبع پروژه:

<https://github.com/iliyalesani/ShamsipourAR>

دریافت موتور Unity3D:

<https://unity.com/>

دریافت ابزار توسعه Vuforia:

<https://developer.vuforia.com/downloads/sdk>

دریافت کلید توسعه دهنده Vuforia:

<https://developer.vuforia.com/license-manager>

دریافت نرم افزار Adobe XD:

<https://www.adobe.com/uk/products/xd.html>

دریافت محیط توسعه Visual Studio:

<https://visualstudio.microsoft.com/>

منابع و مآخذ

مطالب مرتبط Vuforia

<https://www.ptc.com/en/products/augmented-reality>

طراحی اسکلت نرم افزار

<https://uxmovement.com/wireframes/why-its-important-to-sketch-before-you-wireframe/>

توسعه در پلتفرم UWP

<https://docs.microsoft.com/en-us/windows/uwp/get-started/universal-application-platform-guide>

واقعیت افزوده و تیم کوک

<https://www.techradar.com/uk/news/apples-tim-cook-ar-has-the-ability-to-amplify-human-performance>

کتاب برنامه نویسی C# در Unity

https://subscription.packtpub.com/book/game_development/9781785287596

مستندات کامل Unity

<https://docs.unity3d.com/Manual/index.html>

منابع برنامه نویسی Unity

<https://docs.unity3d.com/ScriptReference/>

