

مستندات پلاگین Zarinpay نسخه 1.0.0.0

توسعه دهنده: سجاد احمدی نیت | تیر 1404 | گروه صنایع خلاق نسیم مهر | حقوق توسعه دهنده محفوظ است!

سلام! این مستندات به شما کمک می‌کند تا پرداخت درون برنامه زرین‌پال رو به راحتی در پروژه یونیتی خودتون ادغام و استفاده کنید. ما سعی کردیم همه چیز رو ساده و تمیز نگه داریم تا تمرکز شما روی ساخت بازی‌تون باشه، نه درگیری با فرآیندهای پرداخت. استفاده و نشر این پلاگین کاملاً رایگان و آزاده!

فهرست مطالب

1. مقدمه
2. معرفی پلاگین و هدف
3. قابلیت‌ها
4. ساختار پوشه‌ها و اجزای پلاگین
5. آموزش نصب و راه‌اندازی سریع - Quick Start
6. جزئیات و تشریح بخش‌های مهم
 - a. کدهای یونیتی
 - b. کدهای سمت سرور
7. نکات مهم و امنیتی
8. خلاصه قدم به قدم راه‌اندازی
9. کپی‌رایت و مجوز استفاده

1. مقدمه

این پلاگین برای ساده سازی پرداخت IAP بازی یا اپلیکیشن شما توسط زرین‌پال طراحی شده. تمرکز اینجانب امنیت در سطح متوسطه، راحتی و سرعت راه‌اندازی بوده و سعی شده کدها قابل توسعه و تمام نیازهای پایه یک فروشگاه کوچک را پوشش دهد. لطفاً دقت کنید که جهت امن سازی بیشتر، باید مرچنت کد درگاه را در مکانی امن تر سازماندهی کنید و اسکریپت ZarinpalBuy به همین منظور، بصورت آزاد و با قابلیت درک بسیار بالا، قرارداد شده و بصورت یک کتابخانه با فایل dll تعبیه نشده است.

2. معرفی پلاگین و هدف

پلاگین ZarinPay یک راه‌حل راحت و امن برای پرداخت آیتم‌های داخل بازی/اپلیکیشن هست که:

- دو بخش کلاینت یونیتی و سرور PHP دارد
- فقط سرور به مرچنت کد دسترسی دارد (امنیت بالا)

- خروجی مختصر و با قابلیت اتصال آسان به UI بازی دارد

*** این پلاگین صرفاً پرداخت مستقیم و با ارسال و استعلام اطلاعات با استفاده از POST و GET میباشد.

3. قابلیت‌ها

- ✓ راه‌اندازی سریع و راحت
- ✓ امنیت بالا با معماری client-server
- ✓ سازگار با یونیتی UI و دکمه‌ها
- ✓ هر محصول/دکمه فقط با یک پارامتر قابل تعریف است
- ✓ قابلیت افزودن آیتم یا بسته جدید فقط با چند خط کد
- ✓ نمایش پنل‌های نتیجه و وضعیت پرداخت (موفق، ناموفق، تکراری، ...)
- ✓ سازگاری با پکیج فارسی‌ساز متون
- ✓ دارای نمونه صحنه تست آماده

4. ساختار پوشه‌ها و اجزای پلاگین

Assets/

└─ Zarinpal/

| └─ Prefabs/

| | └─ ZarinpalBuy.prefab ← شامل اسکریپت اصلی و راحت برای کشیدن در صحنه

| | └─ Panels.prefab ← پنل‌های آماده مدیریت وضعیت پرداخت

| └─ Scripts/

| | └─ ZarinpalBuy.cs ← کد اصلی مدیریت پرداخت و رفتار پس از آن

| | └─ PersianText/ ... ← (UI جهت سازگاری متون فارسی در) پکیج فارسی‌ساز متون

| └─ Server/

| | └─ check.php ← کد چک کردن تراکنش برای یونیتی (باید بر روی سرور یا هاست بارگذاری شود)

| | └─ verify.php ← کد اتصال یونیتی به درگاه موردنظر (باید بر روی سرور یا هاست بارگذاری شود)

| └─ Test/

| └─ ZarinpalTestScene.unity ← صحنه تست آماده با دکمه‌ها و پنل نمونه

5. آموزش نصب و راه‌اندازی سریع

● آپلود فایل‌های سمت سرور:

- فایل‌های verify.php و check.php را روی هاست یا سرور خودتان آپلود کنید.
- مقدار مرجنت کد را فقط در سرور وارد کنید.
- آدرس‌ها ثابت بماند.

● انتقال پوشه Zarinpay به پروژه Unity

- پوشه Zarinpay را به مسیر Assets منتقل کنید (یا پکیج اون رو import کنید).

● افزودن به صحنه:

- از پوشه Prefabs پری فب ZarinpalBuy را وارد صحنه بکشید.
- در صورت نیاز، Panels را هم برای نمایش وضعیت پرداخت به Canvas اضافه کنید.

● اتصال دکمه‌های خرید:

- هر دکمه خرید را با متد StartPaymentByProduct(int) ست کنید.
- مقدار int آیتم (مثل 1 برای بسته 1 سکه، 2 برای بسته 5 سکه و ...) را ست کنید.

● تنظیم فیلدهای اصلی:

- متن سکه (Coin)، متن وضعیت (StatusText) و پنل‌های وضعیت را در Inspector جایگذاری کنید (در این یک مورد شما علاوه بر دکمه‌ها و رویدادهایی که پیش از خرید و پس از خرید انجام می‌دهند، می‌توانید آنرا تغییر دهید و خودرا منوط به سکه نکنید)

● ساختار سرور:

- آدرس فایل‌های verify.php و check.php در کد یونیتی باید به درستی تنظیم شود (خط 53 و 112 کد ZarinpalBuy که باید لینک مستقیم این دو فایل باشند و از ساختارهای زیر پیروی کنند:
https://yourdomain.ir/verify.php?order_id={orderId}&amount={payAmount}
https://yourdomain.ir/check.php?order_id={orderId}&mark_delivered=1
که معمولا جای yourdomain نام دامنه خود را می‌گذاریم.

نکته: اگر دامنه سایت شما گواهی ssl ندارد، به جای https حتما http بنویسید.

- **نکته:** اگر فایل اسکریپت شما در پوشه/پوشه‌های میانی قرار دارد، حتما مسیر آنرا دخیل کنید. مثلا: https://yourdomain.ir/zarinpay/verify.php?order_id={orderId}&amount={payAmount} که در این بین فایل verify.php در پوشه zarinpay قرار دارد. فایل php باید به گونه ای باشد که وقتی مسیر آنرا در مرورگر copy-paste میکنید، بدون اشکال اجرا شود و پیغامی ننویسد یا پیغام موفقیت آمیز چاپ کند.

نکته: می‌توانید هاست یا سرور را از میزبانهای وب و سرور دریافت کنید. هاست (host) یک فضای وب است که به شما اجازه می‌دهد فایلها و اسکریپتهای خود را بارگذاری کنید.

- در یونیتی در فیلد merchant_id در اسکریپت ZarinpalBuy باید مرجنت کد خود را بگذارید که می‌توانید آنرا از بخش "تنظیمات درگاه پرداخت" در بخش "کد درگاه پرداخت" کپی کنید.

6. جزئیات و تشریح بخش‌های مهم

متغیرهای مهم و قابل ویرایش ، در اسکریپت ZarinpalBuy :

متغیر	توضیح	قابلیت ویرایش در inspector	جزوی از نمونه یا اسکریپت پایه؟
<code>public Text Coin;</code>	متن رابط کاربری نشان‌دهنده تعداد سکه فعلی	بله	خیر (صرفاً جهت نمونه)
<code>public Text StatusText;</code>	پیام وضعیت نکات پرداخت	بله	خیر (صرفاً جهت نمونه)
<code>int _coin;</code>	سکه عددی کاربر (داخل کد)	خیر	خیر (صرفاً جهت نمونه)
<code>public GameObject _Successfull_panel;</code>	پنل نمایش موفقیت	بله	خیر (صرفاً جهت نمونه)
<code>public GameObject _Paying_panel;</code>	پنل درحال پرداخت	بله	خیر (صرفاً جهت نمونه)
<code>public GameObject _Failed_panel;</code>	پنل عدم موفقیت پرداخت	بله	خیر (صرفاً جهت نمونه)
<code>public GameObject _rec_panel;</code>	پنل درحال فرآیند پرداخت	بله	خیر (صرفاً جهت نمونه)
<code>public GameObject _Unspecified_panel;</code>	پنل اتفاق غیرمشخص در پرداخت	بله	خیر (صرفاً جهت نمونه)
<code>public GameObject _NoConnect_panel;</code>	پنل عدم اتصال به درگاه یا اینترنت	بله	خیر (صرفاً جهت نمونه)
<code>string orderId;</code>	شناسه کاربری منحصر به فرد (تولید خودکار توسط کد)	خیر (اتوماتیک تولید می‌شود)	بله (برای درخواست مهم است)
<code>ShopProductType _productType</code>	نوع محصول خریداری شده	خیر (با خرید ست می‌شود)	بله (برای درخواست مهم است)
<code>string merchant_id;</code>	کد درگاه پرداخت شما (بهتر است در حالت تست یا sandbox نباشد)	خیر	بله (برای درخواست مهم است) (متناسب با درگاه خود ویرایش کنید)
<code>int _lastAmountChecked;</code>	آخرین رقم چک شده (جهت عدم تکرار تراکنش بصورت پشت سرهم)	خیر (توسط اسکریپت پر می‌شود)	بله (برای درخواست مهم است)

شمارنده ای که برای تعیین نوع محصولات مهم است و در حین استفاده از دکمه ها جهت فراخوانی پرداخت، شماره نوع محصول موردنظر را باید از پیش در enum تعریف کرده و عدد آنرا به رویداد دکمه بدهید:

```
public enum ShopProductType
{
    Coin1 = 1,
    Coin5 = 2,
}
```

متدهای اصلی اسکریپت و نقش آن‌ها:

- **StartPaymentByProduct(int type)**

- متد اصلی فراخوانی پرداخت.
- **ورودی:** شماره نوع محصول (enum)
- انجام دهنده تنظیمات اولیه و فراخوانی پرداخت.
- payAmount را بر حسب نوع محصول باید در همین متد ست شود(در case های switch شما میتوانید فعالیت های مختلفی تعریف کنید و یا نسبت به شمارنده هایی که در ShopProductType تعریف کردید، در هر case بیان کنید که اگر case موردنظر باشد {مثلا دکمه ای فشار داده شود که قصد خرید 1 سکه دارد و بطور مثال عدد شمارنده Coin1 را دارد}، چه اتفاقی بیفتد که در این پلاگین بعنوان مثال، سکه را درنظر گرفتیم که 1 سکه(Coin1) مساوی است با هزار تومان). پس برای افزودن محصول جدید، یک مقدار به enum و case جدید در switch اضافه کنید.

• RequestPayment (Coroutine)

- ارتباط با سرور و دریافت لینک پرداخت.
- پیام وضعیت و نمایش پنل پرداخت را مدیریت می کند.
- معمولاً فقط اگر بخواهید توضیح محصول را داینامیک کنید نیاز به ویرایش دارد.

• CheckPaymentStatus و CheckPaymentCoroutine

- برای بررسی وضعیت سفارش و مدیریت اعطای محصول/نمایش پیام استفاده می شود.
- در داخل switch بسته به نوع محصول، پاداش ارائه می شود.
- برای افزودن عملکرد جدید:
 - یک case به switch اضافه کنید.
 - در همان case کد اعطای محصول/ویژگی جدید را بنویسید.

افزودن رویداد وضعیت/محصول جدید

به enum ShopProductType مقدار جدید اضافه کنید. مثلاً:

```
public enum ShopProductType { Coin1 = 1, Coin5 = 2, VipPack = 3, RemoveAds = 4 }
```

که در بالا ما مثلاً VipPack را برای یک پکیج VIP که ربطی به سکه ندارد، و RemoveAds را صرفاً برای حذف تبلیغات قرار دادیم که بخاطر استفاده از آنها در switch، میتوان حالت های مختلف تعریف کرد. دقت کنید که در اسکریپت، هم در StartPaymentByProduct و هم در CheckPaymentCoroutine از شمارنده های ShopProductType برای switch استفاده میشود.

درون switch موجود در StartPaymentByProduct و CheckPaymentCoroutine به ترتیب:

- قیمت جدید و پاداش جدید را تعریف کنید.

- برای پنل ویژه برای حالت/خطای خاص، پنل جدید تعریف کرده و به اسکریپت اضافه کنید.
- دکمه UI جدید بسازید و پارامتر نوع محصول را در Inspector مقدار جدید قرار دهید.

کدهای سمت سرور PHP (verify.php و check.php)

محل تغییر	توضیح	متغیر/فیلد
بالای هر اسکریپت verify و هم check	مرچنت کد زرین پال شما (در هر فایل باید تغییر داده شود. پیشنهاد داده میشود که ابتداء در نوت پد یا VS Code و هر code editor یا text editor دیگری ویرایش شود، سپس فایل آپلود شود و به سی شارپ مسیر داده شود)	\$merchant_id
verify.php	احراز پرداخت (نیاز به ویرایش نیست)	\$authority
verify.php	وضعیت پرداخت (از اسکریپت سی شارپ دریافت میشود)	\$status
verify.php	شماره پرداخت مشتری (از اسکریپت سی شارپ دریافت میشود)	\$order_id
verify.php	پیغام نهایی (نیاز به ویرایش نیست)	\$result_msg
verify.php	متغیر منطقی و بولی "آیا پرداخت صورت گرفته است؟"	\$is_paid

- اسکریپت check.php از فایل json برای ذخیره سازی استفاده میکند.

کدام بخش‌ها اختیاری است؟

- می‌توانید پیام‌های نمایش داده‌شده به کاربر (قسمت HTML در انتهای verify.php) را سفارشی کنید.
- ثبت در دیتابیس و جداکردن کاربر/محصول می‌تواند اختیاری باشد اما شدیداً توصیه به ثبت authority و amount بر اساس userId به صورت امن در دیتابیس می‌شود.
- افزودن محصول جدید به اسکریپت های php معمولاً لازم نیست، چون سمت سرور فقط مبلغ را از دیتابیس می‌خواند (بر اساس authority). مهم است که هر تراکنش با مبلغ و نوع محصول در دیتابیس ذخیره شود.

جایگاه افزودن رویداد یا پنل جدید برای پرداخت

موارد زیر در گذشته بارها تاکید شده است. اما صرفاً جهت یادآوری بیان میشود:

- برای هر محصول یا رویداد، مقدار جدید به enum اضافه می‌کنید.
- case جدید در switch هم در متد آغاز پرداخت (StartPaymentByProduct) هم پس از تایید اعتبار در یونیتی (CheckPaymentCoroutine) ایجاد کنید.
- ساخت پنل جدید (اختیاری):

- اگر بخواهید نتیجه یا وضعیت خاصی را نمایش دهید (مانند پنل وضعیت نامشخص و ...) می‌توانید یک GameObject جدید در Inspector اضافه و آن را در کد اسکریپت صدا بزنید.
- برای سناریوهای خاص (مثلاً تشویق یا هشدار مخصوص) در switch مربوطه پیام یا عملکرد ویژه خود را اضافه کنید.

در نهایت چه چیزهایی باید تغییر داده شود؟

- مرچنت کد. هم در فایل‌های php و هم در فایل سی شارپ. در متغیر merchant_id محتوای بین "" را پاک کرده و مرچنت (کد درگاه پرداخت) خود را جایگذاری کنید.
- enum تعریف شده در ZarinpalBuy.cs که طبق محصولات موردنظر خودتان شمارنده تعریف میشود.
- case های بلوکهای switch در StartPaymentByProduct و CheckPaymentCoroutine در اسکریپت ZarinpalBuy.cs
- متغیر محلی callback_url خط 53 در اسکریپت ZarinpalBuy.cs که تا بخشی از اسکریپت، باید حاوی مسیر مستقیم فایل verify.php باشد.

```
string url = $"https://yourdomain.ir/check.php?order_id={orderId}&mark_delivered=1";
```

- متغیر محلی url خط 112 در اسکریپت ZarinpalBuy.cs که تا بخشی از اسکریپت، باید حاوی مسیر مستقیم فایل check.php باشد.

```
string url = $"https://yourdomain.ir/check.php?order_id={orderId}&mark_delivered=1";
```

و ... سایر فیلدها و متغیرهای GameData که میتوانید به آن افزوده یا از آن کسر کنید.

7. نکات مهم و امنیتی

- **حتماً در صورت تسلط به سی شارپ، مرچنت کد خود را در بازی قرار ندهید. فقط روی سرور! پلاگین** در حد بالا ساده شده است ولی سعی شده امنیت ابتدایی حفظ شود.
- بهتر است callback_url یا اطلاعات اصلی از سرور خوانده شود (برای امنیت بالا).
- بررسی کنید محصول تنها یکبار تحویل داده شود (mark_delivered=1).
- ارورها را در سرور و بازی لاگ بگیرید.

8. خلاصه قدم به قدم راه اندازی

1. ثبت پرونده و حداقل یکی از حسابها، در درگاه مالیاتی
2. ثبت نام در درگاه زرین پال و درخواست درگاه
3. خرید host یا server از میزبان های هاست
4. آپلود فایل های پوشه Server و جایگذاری متغیر مرچنت کد در هردو اسکریپت
5. وارد کردن پکیج یا پوشه Zarinpay در Unity Assets
6. اضافه کردن Prefab ها و تنظیم اسکریپت در صحنه (یا اضافه کردن بصورت دستی)
7. تنظیم اتصال UI به متد پرداخت (با پارامتر)
8. بررسی نتیجه پرداخت و تست روی دستگاه واقعی

*** چنین روش پرداختی، حتی در اندروید هم پاسخگو است. اما در این روش، فرد نیازمند آن است که از مرورگر خارج شده و وارد بازی شود و گزینه "بررسی پرداخت" را کلیک کند(که میتوانید همه آنرا شخصی سازی کنید).

آموزش ثبت درگاه در زرین پال: <https://www.aparat.com/v/mpr2nd7>

آموزش کار با همین پلاگین(ضبط شده توسط خودم): gamescience.ir/product/zarinpay

9. کپی رایت و مجوز استفاده

رفقا حقیقتا جایی ثبتش نکردم و صرفا مستقیما در github قرار دادم. اما چون مرام داریم، بدونید کلیه حقوق متعلق به اینجانب است و قطعا همگی روی گداهایمان خیلی وقت میگذاریم و در طول زمان تحولات بسیاری داشته و وقت گیر است. اما استفاده و انتشار این پلاگین جهت استفاده شخصی رایگان است. اما هرگونه ویرایش و استفاده تجاری، با ذکر منبع پسندیده است.

زندگی آن کس از همه بهتر است که مردم در پرتو احسان و کمک او زندگی کنند. امیرالمومنین علی(علیه السلام)

منبع: میزان الحکمه، جلد ۸، صفحه ۳۴۵

موفق و موید باشید. امیدوارم این مستندات و فایل های همراه آن، تونسته باشه تا به شما کمک کنه. اگر دوست داشتید، میتونید از لینک زرین پالم حمایت کنید:

<https://zarinp.al/nasimmehrgroup>

در کانال شخصی تلگرامم هم میتونید عضو بشید:

https://t.me/Sajjad_Ahmadi_Niat