

```
local DataStoreService = game:GetService("DataStoreService")  
local Players = game:GetService("Players")  
local ReplicatedStorage = game:GetService("ReplicatedStorage")
```

```
local InvestEvent = ReplicatedStorage:WaitForChild("InvestEvent")
```

-- دیتاستور برای ذخیره سرمایه‌گذاری فعال (اختیاری، برای پایداری بین سرورها)

```
local InvestStore = DataStoreService:GetDataStore("InvestmentsV1")
```

-- تنظیمات

```
local PLANS = {  
  LowRisk = {  
    Name = "کم ریسک",  
    MinAmount = 100,  
    Time = 60, -- ثانیه  
    ProfitPercent = 10, -- سود 10%  
    FailChance = 5 -- احتمال از دست دادن کل سرمایه 5%  
  },  
  MediumRisk = {  
    Name = "متوسط",  
    MinAmount = 500,  
    Time = 120,  
    ProfitPercent = 30,  
    FailChance = 20  
  },  
  HighRisk = {  
    Name = "زیاد",  
    MinAmount = 2000,  
    Time = 300,  
    ProfitPercent = 80,  
    FailChance = 45  
  }  
}
```

-- فرض می‌کنیم شما یک سیستم لیدربرد یا اینتولیو به نام "Gold" دارید
-- این تابع باید طبق سیستم ارزی شما شخصی‌سازی شود

```
local function getGold(player)
```

```
local leaderstats = player:FindFirstChild("leaderstats")
if leaderstats then
    local gold = leaderstats:FindFirstChild("Gold")
    if gold then
        return gold
    end
end
return nil
end
```

```
local function addGold(player, amount)
    local gold = getGold(player)
    if gold then
        gold.Value = gold.Value + amount
    end
end
```

```
local function deductGold(player, amount)
    local gold = getGold(player)
    if gold then
        gold.Value = math.max(0, gold.Value - amount)
        return true
    end
    return false
end
```

-- جدول ذخیره موقت سرمایه‌گذاری‌های فعال

```
local ActiveInvestments = {}
```

-- لود کردن داده‌های قبلی (در صورت نیاز)

```
local function loadData(player)
    local userId = player.UserId
    local success, data = pcall(function()
        return InvestStore.GetAsync(userId)
    end)
    if success and data then
        ActiveInvestments[userId] = data
    end
end
```

end

```
local function saveData(player)
    local userId = player.UserId
    if ActiveInvestments[userId] then
        pcall(function()
            InvestStore:SetAsync(userId, ActiveInvestments[userId])
        end)
    end
end
end
```

-- وقتی بازیکن سرمایه گذاری می کند

```
local function onInvest(player, planName)
    local userId = player.UserId
    local plan = PLANS[planName]

    if not plan then
        return "پلن نامعتبر است"
    end

    local goldObj = getGold(player)
    local currentGold = goldObj.Value

    if currentGold < plan.MinAmount then
        return "طلا نیاز دارید " .. plan.MinAmount .. " شما به حداقل"
    end
```

-- چک کردن سرمایه گذاری همزمان (اختیاری: فقط یک سرمایه گذاری فعال)

```
if ActiveInvestments[userId] then
    return "شما در حال حاضر یک سرمایه گذاری فعال دارید"
end
```

-- کسر طلا

```
if not deductGold(player, plan.MinAmount) then
    return "خطا در برداشت طلا"
end
```

-- ایجاد رکورد سرمایه گذاری

```
local investment = {  
    Plan = planName,  
    Amount = plan.MinAmount,  
    StartTime = os.time(),  
    EndTime = os.time() + plan.Time  
}
```

```
ActiveInvestments[userId] = investment  
saveData(player)
```

-- راه اندازی تایمر برای بازپرداخت

```
task.spawn(function()  
    wait(plan.Time)
```

-- چک کردن اگر هنوز سرمایه گذاری پابرجاست

```
if ActiveInvestments[userId] and ActiveInvestments[userId].StartTime ==  
investment.StartTime then
```

```
    local rand = math.random(1, 100)
```

```
    if rand <= plan.FailChance then
```

-- شکست: هیچی پس نمی گیره (کل پول از دست رفته)

-- می تونیم پیام بدیم

```
        local msg = "شکست خورد! شما .. plan.Name .. " سرمایه گذاری شما در پلن "  
plan.MinAmount .. " طلا از دست دادید "
```

-- ارسال پیام به کلاینت (اختیاری)

```
    else
```

-- موفقیت: اصل پول + سود

```
        local profit = math.floor(plan.MinAmount * (plan.ProfitPercent / 100))
```

```
        local totalReturn = plan.MinAmount + profit
```

```
        addGold(player, totalReturn)
```

```
        local msg = "طلا دریافت کردید .. totalReturn .. " سرمایه گذاری موفق! شما "
```

```
(profit .. " سود: ")
```

-- ارسال پیام

```
    end
```

```
ActiveInvestments[userId] = nil
```

```
saveData(player)
```

```
end
```

```
end)

return "سرمايه‌گذاري كرديد " .. plan.Name .. " طلا در پلن " .. plan.MinAmount .. " شما"
end
```

-- هندل ريموت

```
InvestEvent.OnServerInvoke = function(player, action, data)
  if action == "Invest" then
    local planName = data
    local result = onInvest(player, planName)
    return result
  elseif action == "CheckStatus" then
    local userId = player.UserId
    if ActiveInvestments[userId] then
      local inv = ActiveInvestments[userId]
      local remaining = math.max(0, inv.EndTime - os.time())
      return {
        Active = true,
        Plan = PLANS[inv.Plan].Name,
        Amount = inv.Amount,
        RemainingTime = remaining
      }
    else
      return { Active = false }
    end
  end
end
end
```

-- متصل شدن بازيکن

```
Players.PlayerAdded:Connect(function(player)
  loadData(player)
end)
```

-- قطع شدن بازيکن

```
Players.PlayerRemoving:Connect(function(player)
  saveData(player)
  ActiveInvestments[player.UserId] = nil -- پاک کردن از حافظه
end)
```