

Original DLL was the FM and EMV patched version of Redbox.Rental.Services.dll released by Loki on 10/27/2024

<https://discord.com/channels/1292310422749319209/1294293050553667594/1300295294436507678>

Redbox.Rental.Services.KioskProduct > TitleProduct > IsRentable

#All Games were set not Rentable

```
// (get) Token: 0x0600E20 RID: 3616 RVA: 0x000085E0 File Offset: 0x000067E0
public bool IsRentable
{
    get
    {
        return this.IsEmptyCase || (!this.DoNotRent && this.IsValidProduct && this.TitleType !=
            TitleType.DigitalCode && this.TitleFamily != TitleFamily.Games && this.IsAvailable);
    }
}
```

```
// (get) Token: 0x0600E20 RID: 3616 RVA: 0x000085E0 File Offset: 0x000067E0
public bool IsRentable
{
    get
    {
        return this.IsEmptyCase || (!this.DoNotRent && this.IsValidProduct && this.TitleType !=
            TitleType.DigitalCode && this.IsAvailable);
    }
}
```

Redbox.Rental.Services.KioskProduct > TitleProduct > IsSellable

#All games were setr not sellable and all sell price had to be over 0

```
// (get) Token: 0x0600E1E RID: 3614 RVA: 0x000085AF File Offset: 0x000067AF
public bool IsSellable
{
    get
    {
        return this.IsEmptyCase || (this.TitleFamily != TitleFamily.Games &&
            this.PriceEngineIsSellable && this.PurchasePrice > 0m);
    }
}
```

```
// (get) Token: 0x0600E1E RID: 3614 RVA: 0x000085AF File Offset: 0x000067AF
public bool IsSellable
{
    get
    {
        return this.IsEmptyCase || ( this.PriceEngineIsSellable );
    }
}
```

Redbox.Rental.Services.KioskProduct > TitleProduct > DoNotRentDate

#Stopped returning a do not rent date for any title

```
// (get) Token: 0x0000E0 RID: 3593 RVA: 0x000375C8 File Offset: 0x000375C8
public DateTime? DoNotRentDate
{
    get
    {
        return this._title.DoNotRentDate;
    }
}
```

```
public DateTime? DoNotRentDate
{
    get
    {
        return null;
    }
}
```

Redbox.Rental.Services.KioskProduct > TitleProduct > MerchandiseDate

any title has to be merchandised to be available -forcing date to 30 days ago

```
// (get) Token: 0x06000E0E RID: 3598 RVA: 0x000375C8 File Offset: 0x000375C8
public DateTime MerchandiseDate
{
    get
    {
        DateTime? merchandiseDate = this._title.MerchandiseDate;
        if (merchandiseDate == null)
        {
            return DateTime.Now.AddYears(10);
        }
        return merchandiseDate.GetValueOrDefault();
    }
}
```

```
public DateTime MerchandiseDate
{
    get
    {
        return DateTime.Now.AddDays(-30);
    }
}
```

Redbox.Rental.Services.Controllers > BrowseViewController

#Adding a navigation button to be able to view the Games Title Family

#Had to remove Top20 and resize buttons to make room

```
6
7
8 bool flag6 = PersonalizationFunctions.IsSignedIn && RedboxPlusFunctions.CanShowRedboxPlusRedemption;
9 bool isDeselectable = !BaseViewController.IsADAMode;
10 int width = OrchestrationService.IsCultureSpanish() ? 142 : 121;
11 BrowseMenuButton item = BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Top20Movies), highlightSelectedGenre, width, isDeselectable);
12 topMenuModel.MenuButtons.AddRange(new List<BrowseMenuButton>
13 {
14     BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.ActionAndAdventure), highlightSelectedGenre, flag6 ? 101 : 121, isDeselectable),
15     BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Comedy), highlightSelectedGenre, flag6 ? 101 : 121, isDeselectable),
16     BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.FamilyMovies), highlightSelectedGenre, flag6 ? 101 : 121, isDeselectable),
17     BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Horror), highlightSelectedGenre, flag6 ? 101 : 121, isDeselectable),
18     BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Drama), highlightSelectedGenre, flag6 ? 101 : 121, isDeselectable),
19     item,
20     BrowseViewController.CreateGenreMenuButton(null, highlightSelectedGenre, OrchestrationService.IsCultureSpanish() ? 121 : 142, false)
21 });
22 if (flag6)
23 {
24     topMenuModel.MenuButtons.Add(BrowseViewController.CreateSignOutMenuButton());
25 }
26 return;
```

```
Return;
}
bool flag6 = PersonalizationFunctions.IsSignedIn && RedboxPlusFunctions.CanShowRedboxPlusRedemption;
bool isDeselectable = !BaseViewController.IsADAMode;
OrchestrationService.IsCultureSpanish();
topMenuModel.MenuButtons.AddRange(new List<BrowseMenuButton>
{
    BrowseViewController.CreateProductFamilyMenuButton(TitleFamily.Games, ArrowDirection.Left),
    BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.ActionAndAdventure), highlightSelectedGenre, flag6 ? 101 : 111, isDeselectable),
    BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Comedy), highlightSelectedGenre, flag6 ? 101 : 111, isDeselectable),
    BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Horror), highlightSelectedGenre, flag6 ? 101 : 111, isDeselectable),
    BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.FamilyMovies), highlightSelectedGenre, flag6 ? 101 : 111, isDeselectable),
    BrowseViewController.CreateGenreMenuButton(new Genres?(Genres.Drama), highlightSelectedGenre, flag6 ? 101 : 111, isDeselectable),
    BrowseViewController.CreateGenreMenuButton(null, highlightSelectedGenre, OrchestrationService.IsCultureSpanish() ? 121 : 132, false)
});
if (flag6)
{
    topMenuModel.MenuButtons.Add(BrowseViewController.CreateSignOutMenuButton());
}
return;
```

