

Reporting Kit

Reporting Kit provides a lightweight plug-and-play solution to add feedback and bug reporting capabilities directly to your application. It's the one-stop solution for all your in-app reporting needs.

Key Features

- Requires ZERO coding knowledge. Just install, configure and you are ready to use. It's that simple!
- Supports the most popular collaboration tools. We support Jira and Trello services.
- Create different types of reports using a single tool. Supported types are: Bug, Improvement, New Feature, Task, etc
- Share multiple attachments with no restrictions on the file formats.
- Built-in log tracking support enables developers to get more insight into the report.
- Add user-specific information to the report.
- Fully customizable reporting window. You can change the appearance of the composers as per your preference.
- Add new capabilities by using our easily extensible system. Currently, it includes inbuilt support for Essential Kit (add attachments from the mobile gallery) and ReplayKit (add screen recordings).
- Platform agnostic solution. You can use services on any platform.

Supported Services

[Jira](#) | [Trello](#)

Important links

[Forum](#) | [Tutorial](#) | [Support](#)

Namespace **VoxelBusters.ReportingKit**

Classes

ApplicationInfo

Class provides access to application run-time data.

DefaultAssetDataProvider

This class contains default implementation of [IAssetDataProvider](#) interface.

Report

This class contains the basic information about the report.

ReportDataProvider

This class contains default implementation of [IReportDataProvider](#) interface.

ReportingManager

Provides an unified interface to interact with reporting service operations.

SystemInfo

Class provides access to system and hardware information.

Interfaces

IAssetDataProvider

Interface used by the Reporting system to create required asset objects.

IReportCustomDataProvider

Interface provides api's to access app-specific custom information.

IReportDataProvider

Interface provides api's to access information useful to generate [Report](#).

IReportFormatter

Interface provides common implementation for formatting [Report](#).

IReportingService

Implement this interface to support the common operations used by the reporting system.

Enums

BugSeverityLevel

An issue's severity indicates its relative importance.

ReportingServiceType

Defines all the supported reporting service types.

ReportPriorityLevel

Constants indicating the relative importance of an incident and is usually based on the impact and urgency of an issue.

ReportType

Constants indicating the type of [Report](#).

Class ApplicationInfo

Class provides access to application run-time data.

Inheritance

System.Object

ApplicationInfo

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
[Serializable]
public sealed class ApplicationInfo
```

Constructors

ApplicationInfo(ApplicationInfoFieldOptions)

Declaration

```
public ApplicationInfo(ApplicationInfoFieldOptions requiredFields)
```

Parameters

TYPE	NAME	DESCRIPTION
VoxelBusters.ReportingKit.ApplicationInfoFieldOptions	requiredFields	

Properties

FullScreen

Determines whether application is running on full-screen mode (Read Only).

Declaration

```
public bool? FullScreen { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<System.Boolean>	

Identifier

The application identifier at runtime (Read Only). On Apple platforms this is the 'bundleIdentifier' saved in the info.plist file, on Android it's the 'package' from the AndroidManifest.xml.

Declaration

```
public string Identifier { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

Platform

The platform the game is running on (Read Only).

Declaration

```
public RuntimePlatform? Platform { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<RuntimePlatform>	

ScreenOrientation

The logical orientation of the screen (Read Only).

Declaration

```
public ScreenOrientation? ScreenOrientation { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<ScreenOrientation>	

ScreenResolution

The resolution of the screen (Read Only).

Declaration

```
public Resolution? ScreenResolution { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<Resolution>	

UnityVersion

The version of the Unity runtime used to play the content (Read Only).

Declaration

```
public string UnityVersion { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

Version

The application version number (Read Only).

Declaration

```
public string Version { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

Enum BugSeverityLevel

An issue’s severity indicates its relative importance.

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public enum BugSeverityLevel
```

Fields

NAME	DESCRIPTION
Blocker	Indicates that this issue takes precedence over all others (Highest priority).
Critical	Indicates that this issue is causing a problem and requires urgent attention.
Major	Indicates that this issue has a significant impact.
Minor	Indicates that this issue has a relatively minor impact.
Trivial	Indicates that this issue has a lowest priority.
Undefined	Undefined.

Class DefaultAssetDataProvider

This class contains default implementation of [IAssetDataProvider](#) interface.

Inheritance

System.Object
DefaultAssetDataProvider

Implements

[IAssetDataProvider](#)

Namespace: **VoxelBusters.ReportingKit**

Assembly: cs.temp.dll.dll

Syntax

```
public class DefaultAssetDataProvider : IAssetDataProvider
```

Methods

CaptureScreenshot(EventCallback<Asset>)

Declaration

```
public virtual void CaptureScreenshot(EventCallback<Asset> completionCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
EventCallback<Asset>	completionCallback	

GetUnityLogs()

Declaration

```
public virtual Asset GetUnityLogs()
```

Returns

TYPE	DESCRIPTION
Asset	

LoadAssetAtPath(String, String, EventCallback<Asset>)

Declaration

```
public virtual void LoadAssetAtPath(string path, string mimeType, EventCallback<Asset> completionCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	
System.String	mimeType	
EventCallback<Asset>	completionCallback	

OpenFile(String, EventCallback<Asset>)

Declaration

```
public virtual void OpenFile(string path, EventCallback<Asset> completionCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	
EventCallback<Asset>	completionCallback	

RecordScreen(EventCallback<Asset>)

Declaration

```
public virtual void RecordScreen(EventCallback<Asset> completionCallback)
```

Parameters

TYPE	NAME	DESCRIPTION
EventCallback<Asset>	completionCallback	

Implements

[IAssetDataProvider](#)

Interface IAssetDataProvider

Interface used by the Reporting system to create required asset objects.

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public interface IAssetDataProvider
```

Methods

CaptureScreenshot(EventCallback<Asset>)

Captures a screenshot.

Declaration

```
void CaptureScreenshot(EventCallback<Asset> callback)
```

Parameters

TYPE	NAME	DESCRIPTION
EventCallback<Asset>	callback	The callback handler function.

GetUnityLogs()

Returns the cached unity logs as asset file.

Declaration

```
CoreLibrary.Asset GetUnityLogs()
```

Returns

TYPE	DESCRIPTION
CoreLibrary.Asset	The log file.

LoadAssetAtPath(String, String, EventCallback<Asset>)

Loads an asset object available at the specified path.

Declaration

```
void LoadAssetAtPath(string path, string mimeType, EventCallback<Asset> callback)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	Path to the file.
System.String	mimeType	MIME type.

TYPE	NAME	DESCRIPTION
EventCallback<Asset>	callback	The callback handler function.

OpenFile(String, EventCallback<Asset>)

Opens native file browser for a file.

Declaration

```
void OpenFile(string path, EventCallback<Asset> callback)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	Root directory
EventCallback<Asset>	callback	The callback handler function.

RecordScreen(EventCallback<Asset>)

Records the screen activities.

Declaration

```
void RecordScreen(EventCallback<Asset> callback)
```

Parameters

TYPE	NAME	DESCRIPTION
EventCallback<Asset>	callback	The callback handler function.

Interface IReportCustomDataProvider

Interface provides api's to access app-specific custom information.

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public interface IReportCustomDataProvider
```

Methods

GetAdditionalInfo()

Returns the application specific custom information.

Declaration

```
Dictionary<string, string> GetAdditionalInfo()
```

Returns

TYPE	DESCRIPTION
Dictionary<System.String, System.String>	Collection of custom information.

Interface IReportDataProvider

Interface provides api's to access information useful to generate [Report](#).

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public interface IReportDataProvider
```

Methods

GetAdditionalInfo()

Returns the application specific custom information.

Declaration

```
Dictionary<string, string> GetAdditionalInfo()
```

Returns

TYPE	DESCRIPTION
Dictionary<System.String, System.String>	Collection of custom information.

GetApplicationInfo()

Returns the runtime application data.

Declaration

```
ApplicationInfo GetApplicationInfo()
```

Returns

TYPE	DESCRIPTION
ApplicationInfo	The runtime application data.

GetSystemInfo()

Returns the system information.

Declaration

```
SystemInfo GetSystemInfo()
```

Returns

TYPE	DESCRIPTION
SystemInfo	The system information.

Interface IReportFormatter

Interface provides common implementation for formatting [Report](#).

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public interface IReportFormatter
```

Methods

FormatDescription(Report)

Returns the formatted description of the specified report.

Declaration

```
string FormatDescription(Report report)
```

Parameters

TYPE	NAME	DESCRIPTION
Report	report	The report.

Returns

TYPE	DESCRIPTION
System.String	The formatted description.

Interface IReportingService

Implement this interface to support the common operations used by the reporting system.

Inherited Members

[IReportFormatter.FormatDescription\(Report\)](#)

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public interface IReportingService : IReportFormatter
```

Properties

ServiceName

Returns the service type.

Declaration

```
string ServiceName { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

ServiceType

Returns the service type.

Declaration

```
ReportingServiceType ServiceType { get; }
```

Property Value

TYPE	DESCRIPTION
ReportingServiceType	

SprintOptions

Returns an array of sprints that are configured within the service. Here, key represents the display name and value represents the unique_id generated by the system.

Declaration

```
ReadableId[] SprintOptions { get; }
```

Property Value

TYPE	DESCRIPTION
ReadableId[]	

Methods

CreateReport(Report, EventCallback<SuccessResponse>)

Creates a new report.

Declaration

```
void CreateReport(Report report, EventCallback<SuccessResponse> callback = null)
```

Parameters

TYPE	NAME	DESCRIPTION
Report	report	The report.
EventCallback<VoxelBusters.ReportingKit.SuccessResponse>	callback	The callback handler function.

CreateReportComposer(IReportDataProvider, IEnumerable<Asset>)

Creates a report composer instance.

Declaration

```
ReportComposer CreateReportComposer(IReportDataProvider dataProvider, IEnumerable<Asset> initialAttachments)
```

Parameters

TYPE	NAME	DESCRIPTION
IReportDataProvider	dataProvider	Data provider.
IEnumerable<Asset>	initialAttachments	An array of attachment.s

Returns

TYPE	DESCRIPTION
ReportComposer	

LoadConfiguration(CompletionCallback)

Loads configuration.

Declaration

```
void LoadConfiguration(CompletionCallback callback = null)
```

Parameters

TYPE	NAME	DESCRIPTION
CompletionCallback	callback	The callback handler function.

Class Report

This class contains the basic information about the report.

Inheritance

System.Object
Report

Namespace: **VoxelBusters.ReportingKit**

Assembly: cs.temp.dll.dll

Syntax

```
[Serializable]
public class Report
```

Constructors

Report(String, String, String, ReportType, ReportPriorityLevel, BugSeverityLevel, Asset[], ApplicationInfo, SystemInfo, Dictionary<String, String>, String, String[])

Initializes the object.

Declaration

```
public Report(string sprintId = null, string summary = null, string description = null, ReportType reportType = ReportType.Undefined, ReportPriorityLevel priorityLevel = ReportPriorityLevel.Undefined, BugSeverityLevel bugSeverityLevel = BugSeverityLevel.Undefined, Asset[] attachments = null, ApplicationInfo applicationInfo = null, SystemInfo systemInfo = null, Dictionary<string, string> additionalInfo = null, string reporter = null, string[] labels = null)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	sprintId	
System.String	summary	The name of the report.
System.String	description	The description of the report.
ReportType	reportType	The report type.
ReportPriorityLevel	priorityLevel	The priority level.
BugSeverityLevel	bugSeverityLevel	The severity level of the issue.
Asset[]	attachments	An array of the report attachments.
ApplicationInfo	applicationInfo	The application runtime-data.

TYPE	NAME	DESCRIPTION
SystemInfo	systemInfo	The system information.
Dictionary<System.String, System.String>	additionalInfo	The application specific data.
System.String	reporter	The name of the reporter.
System.String[]	labels	An array of meta information.

Properties

AdditionalInfo

The application specific custom data (Read Only).

Declaration

```
public Dictionary<string, string> AdditionalInfo { get; }
```

Property Value

TYPE	DESCRIPTION
Dictionary<System.String, System.String>	

ApplicationInfo

The basic application information (Read Only).

Declaration

```
public ApplicationInfo ApplicationInfo { get; }
```

Property Value

TYPE	DESCRIPTION
ApplicationInfo	

Attachments

The array of attachments (Read Only).

Declaration

```
public Asset[] Attachments { get; }
```

Property Value

TYPE	DESCRIPTION
Asset[]	

BugSeverityLevel

The severity level of the issue (Read Only).

Declaration

```
public BugSeverityLevel BugSeverityLevel { get; }
```

Property Value

TYPE	DESCRIPTION
BugSeverityLevel	

Description

A detailed description of the report (Read Only).

Declaration

```
public string Description { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

Labels

An array of string values providing meta information about the report.

Declaration

```
public string[] Labels { get; }
```

Property Value

TYPE	DESCRIPTION
System.String[]	

PriorityLevel

The priority level of the report in relation to the other reports (Read Only).

Declaration

```
public ReportPriorityLevel PriorityLevel { get; }
```

Property Value

TYPE	DESCRIPTION
ReportPriorityLevel	

Reporter

The name of the reporter (Read Only).

Declaration

```
public string Reporter { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

ReportType

The type of the report (Read Only).

Declaration

```
public ReportType ReportType { get; }
```

Property Value

TYPE	DESCRIPTION
ReportType	

SprintId

The string that identifies the group within reporting service platform (Read Only).

Declaration

```
public string SprintId { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

Summary

A brief one-line summary of the report (Read Only).

Declaration

```
public string Summary { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

SystemInfo

The system information (Read Only).

Declaration

```
public SystemInfo SystemInfo { get; }
```

Property Value

TYPE	DESCRIPTION
SystemInfo	

Methods

GetFormattedDescription(IReportFormatter)

Returns the formatted description of the report.

Declaration

```
public string GetFormattedDescription(IReportFormatter formatter)
```

Parameters

TYPE	NAME	DESCRIPTION
IReportFormatter	formatter	Formatter object.

Returns

TYPE	DESCRIPTION
System.String	The formatted description.

Class ReportDataProvider

This class contains default implementation of [IReportDataProvider](#) interface.

Inheritance

System.Object
ReportDataProvider

Implements

[IReportDataProvider](#)

Namespace: **VoxelBusters.ReportingKit**

Assembly: cs.temp.dll.dll

Syntax

```
public sealed class ReportDataProvider : IReportDataProvider
```

Constructors

ReportDataProvider(ApplicationInfoFieldOptions, SystemInfoFieldOptions, IReportCustomDataProvider)

Declaration

```
public ReportDataProvider(ApplicationInfoFieldOptions applicationInfoRequiredFields, SystemInfoFieldOptions systemInfoFieldOptions, IReportCustomDataProvider customDataProvider = null)
```

Parameters

TYPE	NAME	DESCRIPTION
VoxelBusters.ReportingKit.ApplicationInfoFieldOptions	applicationInfoRequiredFields	
VoxelBusters.ReportingKit.SystemInfoFieldOptions	systemInfoFieldOptions	
IReportCustomDataProvider	customDataProvider	

Methods

GetAdditionalInfo()

Declaration

```
public Dictionary<string, string> GetAdditionalInfo()
```

Returns

TYPE	DESCRIPTION
Dictionary<System.String, System.String>	

GetApplicationInfo()

Declaration

```
public ApplicationInfo GetApplicationInfo()
```

Returns

TYPE	DESCRIPTION
ApplicationInfo	

TYPE	DESCRIPTION

GetSystemInfo()

Declaration

```
public SystemInfo GetSystemInfo()
```

Returns

TYPE	DESCRIPTION
SystemInfo	

Implements

[IReportDataProvider](#)

Class ReportingManager

Provides an unified interface to interact with reporting service operations.

Inheritance

System.Object

ReportingManager

Namespace: **VoxelBusters.ReportingKit**

Assembly: cs.temp.dll.dll

Syntax

```
public class ReportingManager : PrivateSingletonBehaviour<ReportingManager>
```

Properties

CustomAssetDataProvider

Custom [IAssetDataProvider](#) reference.

Declaration

```
public static IAssetDataProvider CustomAssetDataProvider { get; set; }
```

Property Value

TYPE	DESCRIPTION
IAssetDataProvider	

CustomDataProvider

Custom [IReportCustomDataProvider](#) reference.

Declaration

```
public static IReportCustomDataProvider CustomDataProvider { get; set; }
```

Property Value

TYPE	DESCRIPTION
IReportCustomDataProvider	

Methods

CreateReport(Asset[], EventCallback<Report>)

Creates a new report.

Declaration

```
public static void CreateReport(Asset[] attachments = null, EventCallback<Report> callback = null)
```

Parameters

TYPE	NAME	DESCRIPTION
Asset[]	attachments	Adds the specified data as an attachment.

TYPE	NAME	DESCRIPTION
EventCallback< Report >	callback	The callback handler function.

CreateReportWithFile(String, String, EventCallback<Report>)

Creates a new report.

Declaration

```
public static void CreateReportWithFile(string path, string mimeType, EventCallback<Report> callback = null)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	The path to the attachment.
System.String	mimeType	The MIME type of the specified attachment.
EventCallback< Report >	callback	The callback handler function.

CreateReportWithScreenRecording(EventCallback<Report>)

Creates a new report.

Declaration

```
public static void CreateReportWithScreenRecording(EventCallback<Report> callback = null)
```

Parameters

TYPE	NAME	DESCRIPTION
EventCallback< Report >	callback	The callback handler function.

CreateReportWithScreenShot(EventCallback<Report>)

Creates a new report.

Declaration

```
public static void CreateReportWithScreenShot(EventCallback<Report> callback = null)
```

Parameters

TYPE	NAME	DESCRIPTION
EventCallback< Report >	callback	The callback handler function.

OnSingletonAwake()

Declaration

```
protected override void OnSingletonAwake()
```

SetCustomService(IReportingService)

Sets the reference of the custom reporting service.

Declaration

```
public static void SetCustomService(IReportingService service)
```

Parameters

TYPE	NAME	DESCRIPTION
IReportingService	service	The custom service.

Enum ReportingServiceType

Defines all the supported reporting service types.

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public enum ReportingServiceType
```

Fields

NAME	DESCRIPTION
Custom	Indicates that system will use custom reporting.
Jira	Indicates that system will use Jira service.
Trello	Indicates that system will use Trello service.

Enum ReportPriorityLevel

Constants indicating the relative importance of an incident and is usually based on the impact and urgency of an issue.

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public enum ReportPriorityLevel
```

Fields

NAME	DESCRIPTION
High	Serious problem that could block progress.
Highest	The problem will block progress.
Low	Minor problem or easily worked around.
Lowest	Trivial problem with little or no impact on progress.
Medium	Has the potential to affect progress.
Undefined	Undefined.

Enum ReportType

Constants indicating the type of [Report](#).

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
public enum ReportType
```

Fields

NAME	DESCRIPTION
Bug	Indicates that report contains information about the issue.
Improvement	Indicates that report contains information about the suggested improvement.
NewFeature	Indicates that report contains information about the suggested feature.
Task	Indicates that report contains information about the task.
Undefined	Undefined.

Class SystemInfo

Class provides access to system and hardware information.

Inheritance

System.Object
SystemInfo

Namespace: [VoxelBusters.ReportingKit](#)

Assembly: cs.temp.dll.dll

Syntax

```
[Serializable]  
public sealed class SystemInfo
```

Constructors

SystemInfo(SystemInfoFieldOptions)

Declaration

```
public SystemInfo(SystemInfoFieldOptions requiredFields)
```

Parameters

TYPE	NAME	DESCRIPTION
VoxelBusters.ReportingKit.SystemInfoFieldOptions	requiredFields	

Properties

DeviceModel

The model of the device (Read Only).

Declaration

```
public string DeviceModel { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

DeviceType

The kind of device the application is running on (Read Only).

Declaration

```
public DeviceType? DeviceType { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<DeviceType>	

GraphicsDeviceName

The name of the graphics device (Read Only).

Declaration

```
public string GraphicsDeviceName { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

GraphicsDeviceType

The graphics API type used by the graphics device (Read Only).

Declaration

```
public UnityEngine.Rendering.GraphicsDeviceType? GraphicsDeviceType { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<UnityEngine.Rendering.GraphicsDeviceType>	

GraphicsDeviceVendor

The vendor of the graphics device (Read Only).

Declaration

```
public string GraphicsDeviceVendor { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

GraphicsMemorySize

Amount of video memory present (Read Only).

Declaration

```
public int? GraphicsMemorySize { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<System.Int32>	

GraphicsMultiThreaded

Is graphics device using multi-threaded rendering (Read Only)?

Declaration

```
public bool? GraphicsMultiThreaded { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<System.Boolean>	

MemorySize

Amount of system memory present (Read Only).

Declaration

```
public int? MemorySize { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<System.Int32>	

OperatingSystem

Operating system name with version (Read Only).

Declaration

```
public string OperatingSystem { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	

ProcessorCount

Number of processors present (Read Only).

Declaration

```
public int? ProcessorCount { get; }
```

Property Value

TYPE	DESCRIPTION
System.Nullable<System.Int32>	

ProcessorType

Processor name (Read Only).

Declaration

```
public string ProcessorType { get; }
```

Property Value

TYPE	DESCRIPTION
System.String	