

## Game Core

2.00v

Generated by Doxygen 1.8.10

Fri Jul 31 2015 09:55:31



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Packages	1
<b>2</b>	<b>Hierarchical Index</b>	<b>3</b>
2.1	Class Hierarchy	3
<b>3</b>	<b>Class Index</b>	<b>5</b>
3.1	Class List	5
<b>4</b>	<b>Namespace Documentation</b>	<b>7</b>
4.1	achievement_system Namespace Reference	7
4.2	game_core Namespace Reference	7
<b>5</b>	<b>Class Documentation</b>	<b>11</b>
5.1	achievement_system.achievement Class Reference	11
5.1.1	Detailed Description	11
5.1.2	Member Function Documentation	12
5.1.2.1	Awake()	12
5.1.2.2	Start()	12
5.1.3	Property Documentation	12
5.1.3.1	filter	12
5.1.3.2	isAccomplished	12
5.2	achievement_system.actionBehaviour Class Reference	12
5.2.1	Detailed Description	12
5.3	achievement_system.amountReachedFilter Class Reference	12
5.3.1	Detailed Description	13
5.3.2	Constructor & Destructor Documentation	13
5.3.2.1	amountReachedFilter()	13
5.3.3	Member Function Documentation	13
5.3.3.1	test()	13
5.3.4	Property Documentation	13
5.3.4.1	value	13
5.4	game_core.BarView Class Reference	14

5.4.1	Detailed Description	14
5.4.2	Constructor & Destructor Documentation	14
5.4.2.1	BarView(string name)	14
5.4.3	Property Documentation	14
5.4.3.1	active	14
5.4.3.2	gameObject	15
5.4.3.3	position	15
5.4.3.4	rectTransform	15
5.4.3.5	text	15
5.4.3.6	transform	15
5.4.3.7	value	15
5.5	game_core.BoundaryBehaviour Class Reference	15
5.5.1	Detailed Description	16
5.6	game_core.ButtonBehaviour Class Reference	16
5.6.1	Detailed Description	16
5.6.2	Member Function Documentation	17
5.6.2.1	FixedUpdate()	17
5.6.2.2	OnClickEvent()	17
5.6.2.3	OnEnable()	17
5.6.2.4	OnMouseDown()	17
5.6.2.5	OnMouseDownDrag()	17
5.6.2.6	OnMouseEnter()	17
5.6.2.7	OnMouseExit()	17
5.6.2.8	OnMouseOver()	17
5.6.2.9	OnMouseUp()	17
5.6.2.10	Start()	17
5.6.2.11	Update()	17
5.7	game_core.ButtonView Class Reference	18
5.7.1	Detailed Description	18
5.7.2	Constructor & Destructor Documentation	18
5.7.2.1	ButtonView(string name)	18
5.7.3	Property Documentation	18
5.7.3.1	active	18
5.7.3.2	gameObject	19
5.7.3.3	interactable	19
5.7.3.4	position	19
5.7.3.5	rectTransform	19
5.7.3.6	transform	19
5.8	game_core.CanvasButton Class Reference	19
5.8.1	Detailed Description	20

5.8.2	Member Function Documentation	20
5.8.2.1	action()	20
5.9	game_core.CreditsBehaviour Class Reference	20
5.9.1	Detailed Description	20
5.9.2	Member Function Documentation	21
5.9.2.1	OnClick()	21
5.10	game_core.DestroyOnContact Class Reference	21
5.10.1	Detailed Description	21
5.11	game_core.FadePanelBehaviour Class Reference	21
5.11.1	Detailed Description	22
5.11.2	Member Function Documentation	22
5.11.2.1	fadeOut()	22
5.12	achievement_system.Filter Class Reference	22
5.12.1	Detailed Description	22
5.12.2	Constructor & Destructor Documentation	22
5.12.2.1	Filter()	22
5.12.3	Member Function Documentation	23
5.12.3.1	test()	23
5.13	game_core.FSMSys< T, E > Class Template Reference	23
5.13.1	Detailed Description	23
5.13.2	Member Function Documentation	24
5.13.2.1	AddState(State< T, E > s)	24
5.13.2.2	DeleteState(E id)	24
5.13.2.3	PerformTransition(T trans)	24
5.14	game_core.GameBehaviour Class Reference	24
5.14.1	Detailed Description	24
5.15	game_core.legalBehaviour Class Reference	24
5.15.1	Detailed Description	25
5.16	game_core.LevelManager Class Reference	25
5.16.1	Detailed Description	25
5.16.2	Member Function Documentation	26
5.16.2.1	Load(string name)	26
5.16.3	Property Documentation	27
5.16.3.1	Instance	27
5.16.3.2	loadingLevel	27
5.17	game_core.LinkCanvasButton Class Reference	27
5.17.1	Detailed Description	27
5.17.2	Member Function Documentation	28
5.17.2.1	action()	28
5.18	game_core.loadingScreen Class Reference	28

5.18.1 Detailed Description . . . . .	28
5.18.2 Member Data Documentation . . . . .	28
5.18.2.1 backgroundColor . . . . .	28
5.18.2.2 message . . . . .	28
5.18.2.3 style . . . . .	28
5.18.2.4 textColor . . . . .	29
5.19 achievement_system.missionAccomplishedFilter Class Reference . . . . .	29
5.19.1 Detailed Description . . . . .	29
5.19.2 Constructor & Destructor Documentation . . . . .	29
5.19.2.1 missionAccomplishedFilter() . . . . .	29
5.19.3 Member Function Documentation . . . . .	29
5.19.3.1 test() . . . . .	29
5.20 game_core.MusicButtonBehaviour Class Reference . . . . .	30
5.20.1 Detailed Description . . . . .	30
5.20.2 Member Function Documentation . . . . .	30
5.20.2.1 action() . . . . .	30
5.20.2.2 Start() . . . . .	30
5.21 game_core.MusicMuteBehaviour Class Reference . . . . .	31
5.21.1 Detailed Description . . . . .	31
5.21.2 Member Function Documentation . . . . .	31
5.21.2.1 setActive(bool value) . . . . .	31
5.22 game_core.objectFade Class Reference . . . . .	31
5.22.1 Detailed Description . . . . .	32
5.23 game_core.ObjectPool Class Reference . . . . .	32
5.23.1 Detailed Description . . . . .	32
5.23.2 Member Function Documentation . . . . .	32
5.23.2.1 activateObject() . . . . .	32
5.24 game_core.PanelView Class Reference . . . . .	32
5.24.1 Detailed Description . . . . .	33
5.24.2 Constructor & Destructor Documentation . . . . .	33
5.24.2.1 PanelView(string name) . . . . .	33
5.24.3 Property Documentation . . . . .	33
5.24.3.1 active . . . . .	33
5.24.3.2 gameObject . . . . .	33
5.24.3.3 position . . . . .	33
5.24.3.4 rectTransform . . . . .	34
5.24.3.5 transform . . . . .	34
5.25 game_core.particleSortingLayer Class Reference . . . . .	34
5.25.1 Detailed Description . . . . .	34
5.26 game_core.PauseAudioBehaviour Class Reference . . . . .	34

5.26.1 Detailed Description . . . . .	35
5.27 game_core.PauseButton Class Reference . . . . .	35
5.27.1 Detailed Description . . . . .	35
5.27.2 Member Function Documentation . . . . .	35
5.27.2.1 action() . . . . .	35
5.28 game_core.ScreenShakeBehaviour Class Reference . . . . .	35
5.28.1 Detailed Description . . . . .	36
5.28.2 Member Function Documentation . . . . .	36
5.28.2.1 OnShake() . . . . .	36
5.28.3 Member Data Documentation . . . . .	36
5.28.3.1 camTransform . . . . .	36
5.28.3.2 onEnableShake . . . . .	36
5.28.3.3 shakeAmount . . . . .	36
5.29 game_core.SocialCanvasButton Class Reference . . . . .	37
5.29.1 Detailed Description . . . . .	37
5.29.2 Member Function Documentation . . . . .	37
5.29.2.1 action() . . . . .	37
5.29.2.2 Start() . . . . .	38
5.30 game_core.SoundCanvasButton Class Reference . . . . .	38
5.30.1 Detailed Description . . . . .	38
5.30.2 Member Function Documentation . . . . .	39
5.30.2.1 action() . . . . .	39
5.30.2.2 Start() . . . . .	39
5.31 game_core.SoundView Class Reference . . . . .	39
5.31.1 Detailed Description . . . . .	39
5.31.2 Constructor & Destructor Documentation . . . . .	40
5.31.2.1 SoundView(string name) . . . . .	40
5.31.3 Property Documentation . . . . .	40
5.31.3.1 active . . . . .	40
5.31.3.2 audioSource . . . . .	40
5.31.3.3 gameObject . . . . .	40
5.31.3.4 pitch . . . . .	40
5.31.3.5 play . . . . .	40
5.31.3.6 position . . . . .	40
5.31.3.7 transform . . . . .	40
5.31.3.8 volume . . . . .	40
5.32 game_core.SpawnerBehaviour Class Reference . . . . .	41
5.32.1 Detailed Description . . . . .	41
5.33 game_core.SplashBehaviour Class Reference . . . . .	41
5.33.1 Detailed Description . . . . .	41

5.34	<a href="#">game_core.StarView Class Reference</a>	42
5.34.1	<a href="#">Detailed Description</a>	42
5.34.2	<a href="#">Constructor &amp; Destructor Documentation</a>	42
5.34.2.1	<a href="#">StarView(string name)</a>	42
5.34.3	<a href="#">Property Documentation</a>	42
5.34.3.1	<a href="#">active</a>	42
5.34.3.2	<a href="#">activeStar</a>	42
5.34.3.3	<a href="#">gameObject</a>	43
5.34.3.4	<a href="#">position</a>	43
5.34.3.5	<a href="#">rectTransform</a>	43
5.34.3.6	<a href="#">transform</a>	43
5.35	<a href="#">game_core.State&lt; T, E &gt; Class Template Reference</a>	43
5.35.1	<a href="#">Detailed Description</a>	44
5.35.2	<a href="#">Member Function Documentation</a>	45
5.35.2.1	<a href="#">Act()</a>	45
5.35.2.2	<a href="#">DeleteTransition(T trans)</a>	45
5.35.2.3	<a href="#">DoBeforeEntering()</a>	45
5.35.2.4	<a href="#">DoBeforeLeaving()</a>	45
5.35.2.5	<a href="#">GetOutputState(T trans)</a>	45
5.35.2.6	<a href="#">Reason()</a>	45
5.36	<a href="#">game_core.StatsData Class Reference</a>	45
5.36.1	<a href="#">Detailed Description</a>	46
5.37	<a href="#">game_core.TextView Class Reference</a>	46
5.37.1	<a href="#">Detailed Description</a>	46
5.37.2	<a href="#">Constructor &amp; Destructor Documentation</a>	46
5.37.2.1	<a href="#">TextView(string name)</a>	46
5.37.3	<a href="#">Property Documentation</a>	46
5.37.3.1	<a href="#">active</a>	46
5.37.3.2	<a href="#">gameObject</a>	47
5.37.3.3	<a href="#">position</a>	47
5.37.3.4	<a href="#">rectTransform</a>	47
5.37.3.5	<a href="#">text</a>	47
5.37.3.6	<a href="#">transform</a>	47
5.38	<a href="#">achievement_system.timeOutFilter Class Reference</a>	47
5.38.1	<a href="#">Detailed Description</a>	48
5.38.2	<a href="#">Constructor &amp; Destructor Documentation</a>	48
5.38.2.1	<a href="#">timeOutFilter()</a>	48
5.38.3	<a href="#">Member Function Documentation</a>	48
5.38.3.1	<a href="#">test()</a>	48
5.38.4	<a href="#">Property Documentation</a>	48



---

5.38.4.1	time	48
5.39	game_core.TouchBehaviour Class Reference	48
5.39.1	Detailed Description	49
5.39.2	Member Function Documentation	49
5.39.2.1	OnTouchBegan(Vector3 v)	49
5.39.2.2	OnTouchCanceled(Vector3 v)	49
5.39.2.3	OnTouchEnded(Vector3 v)	49
5.39.2.4	OnTouchMoved(Vector3 v)	49
5.39.2.5	OnTouchStay(Vector3 v)	50
5.39.2.6	Start()	50
5.39.2.7	Update()	50
5.40	game_core.TouchSystem2D Class Reference	50
5.40.1	Detailed Description	50
5.40.2	Member Data Documentation	50
5.40.2.1	touchInputMask	50
5.41	game_core.TouchSystem3D Class Reference	51
5.41.1	Detailed Description	51
5.41.2	Member Data Documentation	51
5.41.2.1	touchInputMask	51
<b>Index</b>		<b>53</b>



# Chapter 1

## Namespace Index

### 1.1 Packages

Here are the packages with brief descriptions (if available):

<a href="#">achievement_system</a>	.....	7
<a href="#">game_core</a>	.....	7



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

game_core.BarView . . . . .	14
game_core.ButtonView . . . . .	18
achievement_system.Filter . . . . .	22
achievement_system.amountReachedFilter . . . . .	12
achievement_system.missionAccomplishedFilter . . . . .	29
achievement_system.timeOutFilter . . . . .	47
game_core.FSMSys< T, E > . . . . .	23
MonoBehaviour	
achievement_system.achievement . . . . .	11
achievement_system.actionBehaviour . . . . .	12
game_core.BoundaryBehaviour . . . . .	15
game_core.ButtonBehaviour . . . . .	16
game_core.CanvasButton . . . . .	19
game_core.LinkCanvasButton . . . . .	27
game_core.MusicButtonBehaviour . . . . .	30
game_core.PauseButton . . . . .	35
game_core.SocialCanvasButton . . . . .	37
game_core.SoundCanvasButton . . . . .	38
game_core.CreditsBehaviour . . . . .	20
game_core.DestroyOnContact . . . . .	21
game_core.FadePanelBehaviour . . . . .	21
game_core.GameBehaviour . . . . .	24
game_core.legalBehaviour . . . . .	24
game_core.LevelManager . . . . .	25
game_core.loadingScreen . . . . .	28
game_core.MusicMuteBehaviour . . . . .	31
game_core.objectFade . . . . .	31
game_core.ObjectPool . . . . .	32
game_core.particleSortingLayer . . . . .	34
game_core.PauseAudioBehaviour . . . . .	34
game_core.ScreenShakeBehaviour . . . . .	35
game_core.SpawnerBehaviour . . . . .	41
game_core.SplashBehaviour . . . . .	41
game_core.StatsData . . . . .	45
game_core.TouchBehaviour . . . . .	48
game_core.TouchSystem2D . . . . .	50
game_core.TouchSystem3D . . . . .	51

game_core.PanelView . . . . .	32
game_core.SoundView . . . . .	39
game_core.StarView . . . . .	42
game_core.State< T, E > . . . . .	43
game_core.TextView . . . . .	46

## Chapter 3

# Class Index

### 3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">achievement_system.achievement</a>	
This class represents the achievement in the Achievement System. Each achievement has a filter to verify the game progress. The filter can be checked every frame to execute an action when achievement is accomplished. . . . .	11
<a href="#">achievement_system.actionBehaviour</a>	
Action TEST behaviour. . . . .	12
<a href="#">achievement_system.amountReachedFilter</a>	
Quantitative filter; it tests whether the desired amount is reached. . . . .	12
<a href="#">game_core.BarView</a>	
Bar view class; Abstraction layer to deal with UI elements(Bar in this case). . . . .	14
<a href="#">game_core.BoundaryBehaviour</a>	
BoundaryBehaviour class defines the behaviour of boundary when a gameObject leaves from game zone. . . . .	15
<a href="#">game_core.ButtonBehaviour</a>	
Button behaviour. . . . .	16
<a href="#">game_core.ButtonView</a>	
Button view class; Abstraction layer to deal with UI elements(Buttons in this case). . . . .	18
<a href="#">game_core.CanvasButton</a>	
Canvas button class. . . . .	19
<a href="#">game_core.CreditsBehaviour</a>	
Credits behaviour class shows credits slowly on credits scene, whether user touches the credits changes the movement flag. . . . .	20
<a href="#">game_core.DestroyOnContact</a>	
DestroyOnContact class defines the behaviour of gameObject when collide with other. . . . .	21
<a href="#">game_core.FadePanelBehaviour</a>	
Fade panel behaviour class; makes that panels (dis)appear on the screen smoothly. . . . .	21
<a href="#">achievement_system.Filter</a>	
Filter abstract class define a set of operations which all objects that implement class must support. . . . .	22
<a href="#">game_core.FSMSys&lt; T, E &gt;</a>	
FSMSystem class represents the Finite State Machine class. It has a List with the States the NPC has and methods to add, delete a state, and to change the current state the Machine is on. . . . .	23
<a href="#">game_core.GameBehaviour</a>	
Game behaviour class is a test system to verify the right operation of code. . . . .	24
<a href="#">game_core.legalBehaviour</a>	
24	
<a href="#">game_core.LevelManager</a>	
Level manager class; Deals with level load transaction. . . . .	25

<a href="#">game_core.LinkCanvasButton</a>	27
<a href="#">game_core.loadingScreen</a>	28
Loading screen. . . . .	
<a href="#">achievement_system.missionAccomplishedFilter</a>	29
Dichotomic filter; it tests whether the mission is accomplished. . . . .	
<a href="#">game_core.MusicButtonBehaviour</a>	30
Music button behaviour class; Enable/Disable music. . . . .	
<a href="#">game_core.MusicMuteBehaviour</a>	31
Music mute configuration. . . . .	
<a href="#">game_core.objectFade</a>	31
Object fade behaviour. . . . .	
<a href="#">game_core.ObjectPool</a>	32
Object pool class creates and manages object instances. . . . .	
<a href="#">game_core.PanelView</a>	32
Panel view class; Abstraction layer to deal with UI elements(Panel in this case). . . . .	
<a href="#">game_core.particleSortingLayer</a>	34
This class sets the particle sorting layer in 2D perspective. . . . .	
<a href="#">game_core.PauseAudioBehaviour</a>	34
Music behaviour on game pause. . . . .	
<a href="#">game_core.PauseButton</a>	35
Pause button. . . . .	
<a href="#">game_core.ScreenShakeBehaviour</a>	35
Screen shake effect behaviour. . . . .	
<a href="#">game_core.SocialCanvasButton</a>	37
<a href="#">game_core.SoundCanvasButton</a>	38
<a href="#">game_core.SoundView</a>	39
Sound view class; Abstraction layer to deal with UI elements(Sound in this case). . . . .	
<a href="#">game_core.SpawnerBehaviour</a>	41
This class spawns objects with a certain force and every X seconds(timeRange). . . . .	
<a href="#">game_core.SplashBehaviour</a>	41
Splash behaviour. . . . .	
<a href="#">game_core.StarView</a>	42
Star view class; Abstraction layer to deal with UI elements(Star in this case). . . . .	
<a href="#">game_core.State&lt; T, E &gt;</a>	43
This class represents the States in the Finite <a href="#">State</a> System. Each state has a Dictionary with pairs (transition-state) showing which state the FSM should be if a transition is fired while this state is the current state. Method Reason is used to determine which transition should be fired . Method Act has the code to perform the actions the NPC is supposed do if it's on this state. . . . .	
<a href="#">game_core.StatsData</a>	45
Stats data class; manages the stats saved in PlayerPrefs. . . . .	
<a href="#">game_core.TextView</a>	46
Text view class; Abstraction layer to deal with UI elements(Text in this case). . . . .	
<a href="#">achievement_system.timeOutFilter</a>	47
time filter; it tests whether the time is over. . . . .	
<a href="#">game_core.TouchBehaviour</a>	48
Touch behaviour class; Not in use, but define a set of operations which all objects that implement class must support (similar <a href="#">ButtonBehaviour</a> class). . . . .	
<a href="#">game_core.TouchSystem2D</a>	50
Touch system2d(Vector2) class; Deals with the hits on the screen. . . . .	
<a href="#">game_core.TouchSystem3D</a>	51
Touch system3d(Vector3) class; Deals with the hits on the screen. . . . .	



## Chapter 4

# Namespace Documentation

### 4.1 achievement\_system Namespace Reference

#### Classes

- class [achievement](#)  
*This class represents the achievement in the Achievement System. Each achievement has a filter to verify the game progress. The filter can be checked every frame to execute an action when achievement is accomplished.*
- class [actionBehaviour](#)  
*Action TEST behaviour.*
- class [amountReachedFilter](#)  
*Quantitative filter; it tests whether the desired amount is reached.*
- class [Filter](#)  
*Filter abstract class define a set of operations which all objects that implement class must support.*
- class [missionAccomplishedFilter](#)  
*Dichotomic filter; it tests whether the mission is accomplished.*
- class [timeOutFilter](#)  
*time filter; it tests whether the time is over.*

### 4.2 game\_core Namespace Reference

#### Classes

- class [BarView](#)  
*Bar view class; Abstraction layer to deal with UI elements(Bar in this case).*
- class [BoundaryBehaviour](#)  
*BoundaryBehaviour class defines the behaviour of boundary when a gameObject leaves from game zone.*
- class [ButtonBehaviour](#)  
*Button behaviour.*
- class [ButtonView](#)  
*Button view class; Abstraction layer to deal with UI elements(Buttons in this case).*
- class [CanvasButton](#)  
*Canvas button class.*
- class [CreditsBehaviour](#)  
*Credits behaviour class shows credits slowly on credits scene, whether user touches the credits changes the movement flag.*
- class [DestroyOnContact](#)

- [DestroyOnContact](#) class defines the behaviour of gameObject when collide with other.*
- class [FadePanelBehaviour](#)

*Fade panel behaviour class; makes that panels (dis)appear on the screen smoothly.*
  - class [FSMSys](#)

*FSMSystem class represents the Finite [State](#) Machine class. It has a List with the States the NPC has and methods to add, delete a state, and to change the current state the Machine is on.*
  - class [GameBehaviour](#)

*Game behaviour class is a test system to verify the right operation of code.*
  - class [legalBehaviour](#)
  - class [LevelManager](#)

*Level manager class; Deals with level load transaction.*
  - class [LinkCanvasButton](#)
  - class [loadingScreen](#)

*Loading screen.*
  - class [MusicButtonBehaviour](#)

*Music button behaviour class; Enable/Disable music.*
  - class [MusicMuteBehaviour](#)

*Music mute configuration.*
  - class [objectFade](#)

*Object fade behaviour.*
  - class [ObjectPool](#)

*Object pool class creates and manages object instances.*
  - class [PanelView](#)

*Panel view class; Abstraction layer to deal with UI elements(Panel in this case).*
  - class [particleSortingLayer](#)

*This class sets the particle sorting layer in 2D perspective.*
  - class [PauseAudioBehaviour](#)

*Music behaviour on game pause.*
  - class [PauseButton](#)

*Pause button.*
  - class [ScreenShakeBehaviour](#)

*Screen shake effect behaviour.*
  - class **SettingsManager**

*Settings manager class; Deals with player settings saved in PlayerPrefs.*
  - class [SocialCanvasButton](#)
  - class [SoundCanvasButton](#)
  - class [SoundView](#)

*Sound view class; Abstraction layer to deal with UI elements(Sound in this case).*
  - class [SpawnerBehaviour](#)

*This class spawns objects with a certain force and every X seconds(timeRange).*
  - class [SplashBehaviour](#)

*Splash behaviour.*
  - class [StarView](#)

*Star view class; Abstraction layer to deal with UI elements(Star in this case).*
  - class [State](#)

*This class represents the States in the Finite [State](#) System. Each state has a Dictionary with pairs (transition-state) showing which state the FSM should be if a transition is fired while this state is the current state. Method Reason is used to determine which transition should be fired . Method Act has the code to perform the actions the NPC is supposed do if it's on this state.*
  - class **StatsController**
  - class [StatsData](#)

*Stats data class; manages the stats saved in PlayerPrefs.*

- class [TextView](#)  
*Text view class; Abstraction layer to deal with UI elements(Text in this case).*
- class **TimeManager**  
*Time manager class; Controls time flow of the game.*
- class [TouchBehaviour](#)  
*Touch behaviour class; Not in use, but define a set of operations which all objects that implement class must support (similar [ButtonBehaviour](#) class).*
- class [TouchSystem2D](#)  
*Touch system2d(Vector2) class; Deals with the hits on the screen.*
- class [TouchSystem3D](#)  
*Touch system3d(Vector3) class; Deals with the hits on the screen.*



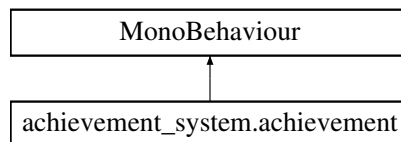
## Chapter 5

# Class Documentation

### 5.1 achievement\_system.achievement Class Reference

This class represents the achievement in the Achievement System. Each achievement has a filter to verify the game progress. The filter can be checked every frame to execute an action when achievement is accomplished.

Inheritance diagram for achievement\_system.achievement:



#### Public Member Functions

- void [Awake](#) ()  
*Use this for initialization.*
- void [Start](#) ()  
*Use this for initialization.*

#### Public Attributes

- float **weight** = 1000.0f
- bool **checkEveryFrame** = false

#### Properties

- bool [isAccomplished](#) [get]  
*Gets a value indicating whether this game\_core.achievement is accomplished.*
- [Filter filter](#) [get, set]  
*Gets or sets the filter.*

#### 5.1.1 Detailed Description

This class represents the achievement in the Achievement System. Each achievement has a filter to verify the game progress. The filter can be checked every frame to execute an action when achievement is accomplished.

### 5.1.2 Member Function Documentation

#### 5.1.2.1 void achievement\_system.achievement.Awake ( )

Use this for initialization.

#### 5.1.2.2 void achievement\_system.achievement.Start ( )

Use this for initialization.

### 5.1.3 Property Documentation

#### 5.1.3.1 Filter achievement\_system.achievement.filter [get], [set]

Gets or sets the filter.

The filter.

#### 5.1.3.2 bool achievement\_system.achievement.isAccomplished [get]

Gets a value indicating whether this game\_core.achievement is accomplished.

true if is accomplished; otherwise, false.

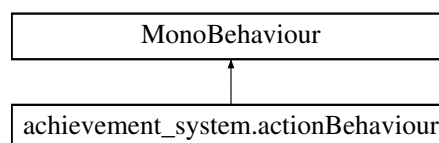
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/achievement↔  
System/achievement.cs

## 5.2 achievement\_system.actionBehaviour Class Reference

Action TEST behaviour.

Inheritance diagram for achievement\_system.actionBehaviour:



### 5.2.1 Detailed Description

Action TEST behaviour.

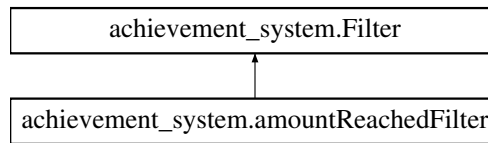
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/achievement↔  
System/actionBehaviour.cs

## 5.3 achievement\_system.amountReachedFilter Class Reference

Quantitative filter; it tests whether the desired amount is reached.

Inheritance diagram for achievement\_system.amountReachedFilter:



## Public Member Functions

- [amountReachedFilter](#) ()  
*Initializes a new instance of the game\_core.timeOutFilter class.*
- override bool [test](#) ()  
*Test this instance.*

## Public Attributes

- float **target** = 0.0f

## Properties

- float [value](#) [get, set]  
*Gets or sets the value.*

### 5.3.1 Detailed Description

Quantitative filter; it tests whether the desired amount is reached.

### 5.3.2 Constructor & Destructor Documentation

#### 5.3.2.1 achievement\_system.amountReachedFilter.amountReachedFilter ( )

Initializes a new instance of the game\_core.timeOutFilter class.

### 5.3.3 Member Function Documentation

#### 5.3.3.1 override bool achievement\_system.amountReachedFilter.test ( ) [virtual]

Test this instance.

Implements [achievement\\_system.Filter](#).

### 5.3.4 Property Documentation

#### 5.3.4.1 float achievement\_system.amountReachedFilter.value [get], [set]

Gets or sets the value.

The value.

The documentation for this class was generated from the following file:

- `/Users/mrferrys/Documents/looneybits/works/unity/game_core/Assets/game_core/scripts/achievementSystem/filters.cs`

## 5.4 game\_core.BarView Class Reference

Bar view class; Abstraction layer to deal with UI elements(Bar in this case).

### Public Member Functions

- `BarView` (string name)  
*Initializes a new instance of the `BarView` class.*

### Properties

- `GameObject` `gameObject` [get]  
*Games the object.*
- `Transform` `transform` [get]  
*Transform this instance.*
- `RectTransform` `rectTransform` [get]  
*Transform this instance.*
- `Vector3` `position` [get]  
*Sets the position.*
- `bool` `active` [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*
- `string` `text` [set]  
*Sets the text.*
- `float` `value` [set]  
*Sets the value.*

#### 5.4.1 Detailed Description

Bar view class; Abstraction layer to deal with UI elements(Bar in this case).

#### 5.4.2 Constructor & Destructor Documentation

##### 5.4.2.1 game\_core.BarView.BarView ( string name )

Initializes a new instance of the `BarView` class.

#### 5.4.3 Property Documentation

##### 5.4.3.1 bool game\_core.BarView.active [get],[set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.



**5.4.3.2** GameObject game\_core.BarView.gameObject [get]

Games the object.

**Returns**

The object.

**5.4.3.3** Vector3 game\_core.BarView.position [get]

Sets the position.

The position.

**5.4.3.4** RectTransform game\_core.BarView.rectTransform [get]

Transform this instance.

**5.4.3.5** string game\_core.BarView.text [set]

Sets the text.

The text.

**5.4.3.6** Transform game\_core.BarView.transform [get]

Transform this instance.

**5.4.3.7** float game\_core.BarView.value [set]

Sets the value.

The value.

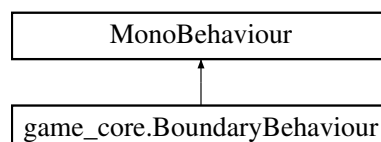
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/views/ui/BarView.cs

**5.5** game\_core.BoundaryBehaviour Class Reference

[BoundaryBehaviour](#) class defines the behaviour of boundary when a gameObject leaves from game zone.

Inheritance diagram for game\_core.BoundaryBehaviour:

**Public Attributes**

- bool **destroy** =true

### 5.5.1 Detailed Description

[BoundaryBehaviour](#) class defines the behaviour of boundary when a gameObject leaves from game zone.

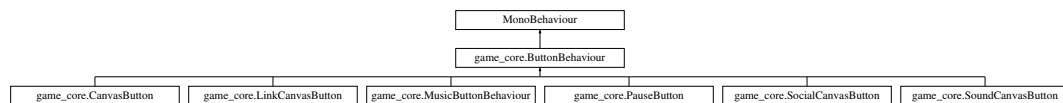
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/BoundaryBehaviour.cs

## 5.6 game\_core.ButtonBehaviour Class Reference

Button behaviour.

Inheritance diagram for game\_core.ButtonBehaviour:



### Public Member Functions

- virtual void [OnClickEvent](#) ()  
*Raises the click event event.*

### Protected Member Functions

- virtual void [Start](#) ()  
*Use this for initialization.*
- virtual void [FixedUpdate](#) ()  
*Fixed Update is called every fixed framerate frame.*
- virtual void [Update](#) ()  
*Update is called once per frame.*
- virtual void [OnEnable](#) ()  
*Raises the enable event.*
- virtual void [OnMouseDown](#) ()  
*Raises the mouse down event.*
- virtual void [OnMouseDownDrag](#) ()  
*Raises the mouse drag event.*
- virtual void [OnMouseEnter](#) ()  
*Raises the mouse enter event.*
- virtual void [OnMouseExit](#) ()  
*Raises the mouse exit event.*
- virtual void [OnMouseOver](#) ()  
*Raises the mouse over event.*
- virtual void [OnMouseUp](#) ()  
*Raises the mouse up event.*
- virtual void **action** ()

### 5.6.1 Detailed Description

Button behaviour.

## 5.6.2 Member Function Documentation

**5.6.2.1** virtual void game\_core.ButtonBehaviour.FixedUpdate ( ) [protected],[virtual]

Fixed Update is called every fixed framerate frame.

**5.6.2.2** virtual void game\_core.ButtonBehaviour.OnClickEvent ( ) [virtual]

Raises the click event event.

**5.6.2.3** virtual void game\_core.ButtonBehaviour.OnEnable ( ) [protected],[virtual]

Raises the enable event.

**5.6.2.4** virtual void game\_core.ButtonBehaviour.OnMouseDown ( ) [protected],[virtual]

Raises the mouse down event.

**5.6.2.5** virtual void game\_core.ButtonBehaviour.OnMouseDown ( ) [protected],[virtual]

Raises the mouse drag event.

**5.6.2.6** virtual void game\_core.ButtonBehaviour.OnMouseEnter ( ) [protected],[virtual]

Raises the mouse enter event.

**5.6.2.7** virtual void game\_core.ButtonBehaviour.OnMouseExit ( ) [protected],[virtual]

Raises the mouse exit event.

**5.6.2.8** virtual void game\_core.ButtonBehaviour.OnMouseOver ( ) [protected],[virtual]

Raises the mouse over event.

**5.6.2.9** virtual void game\_core.ButtonBehaviour.OnMouseUp ( ) [protected],[virtual]

Raises the mouse up event.

**5.6.2.10** virtual void game\_core.ButtonBehaviour.Start ( ) [protected],[virtual]

Use this for initialization.

Reimplemented in [game\\_core.SocialCanvasButton](#), [game\\_core.SoundCanvasButton](#), and [game\\_core.Music↔ButtonBehaviour](#).

**5.6.2.11** virtual void game\_core.ButtonBehaviour.Update ( ) [protected],[virtual]

Update is called once per frame.

The documentation for this class was generated from the following file:

- [/Users/mrferrys/Documents/looneybits/works/unity/game\\_core/Assets/game\\_core/scripts/behaviour/ui/ButtonBehaviour.cs](#)

## 5.7 game\_core.ButtonView Class Reference

Button view class; Abstraction layer to deal with UI elements(Buttons in this case).

### Public Member Functions

- [ButtonView](#) (string name)  
*Initializes a new instance of the [BarView](#) class.*

### Properties

- GameObject [gameObject](#) [get]  
*Games the object.*
- Transform [transform](#) [get]  
*Transform this instance.*
- RectTransform [rectTransform](#) [get]  
*Transform this instance.*
- Vector3 [position](#) [get]  
*Sets the position.*
- bool [active](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*
- bool [interactable](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*

#### 5.7.1 Detailed Description

Button view class; Abstraction layer to deal with UI elements(Buttons in this case).

#### 5.7.2 Constructor & Destructor Documentation

##### 5.7.2.1 game\_core.ButtonView.ButtonView ( string name )

Initializes a new instance of the [BarView](#) class.

#### 5.7.3 Property Documentation

##### 5.7.3.1 bool game\_core.ButtonView.active [get], [set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.

## 5.7.3.2 GameObject game\_core.ButtonView.gameObject [get]

Games the object.

## Returns

The object.

## 5.7.3.3 bool game\_core.ButtonView.interactable [get], [set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.

## 5.7.3.4 Vector3 game\_core.ButtonView.position [get]

Sets the position.

The position.

## 5.7.3.5 RectTransform game\_core.ButtonView.rectTransform [get]

Transform this instance.

## 5.7.3.6 Transform game\_core.ButtonView.transform [get]

Transform this instance.

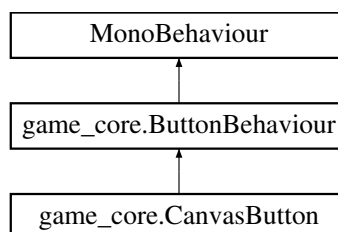
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/views/ui/ButtonView.cs

## 5.8 game\_core.CanvasButton Class Reference

Canvas button class.

Inheritance diagram for game\_core.CanvasButton:



## Public Attributes

- bool **saveld** = false
- string **sceneName** = "menu"
- int **buttonID** = 0

## Protected Member Functions

- override void [action](#) ()  
*Raises the mouse down event.*

## Additional Inherited Members

### 5.8.1 Detailed Description

Canvas button class.

### 5.8.2 Member Function Documentation

#### 5.8.2.1 override void `game_core.CanvasButton.action ( )` [protected],[virtual]

Raises the mouse down event.

Reimplemented from [game\\_core.ButtonBehaviour](#).

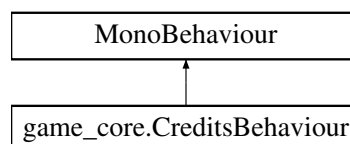
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/Canvas↔  
Button.cs

## 5.9 `game_core.CreditsBehaviour` Class Reference

Credits behaviour class shows credits slowly on credits scene, whether user touches the credits changes the movement flag.

Inheritance diagram for `game_core.CreditsBehaviour`:



## Public Member Functions

- void [OnClick](#) ()  
*Raises the click event.*

## Public Attributes

- float **endMarker**
- float **speed** = 1.0f

### 5.9.1 Detailed Description

Credits behaviour class shows credits slowly on credits scene, whether user touches the credits changes the movement flag.

## 5.9.2 Member Function Documentation

### 5.9.2.1 void game\_core.CreditsBehaviour.OnClick ( )

Raises the click event.

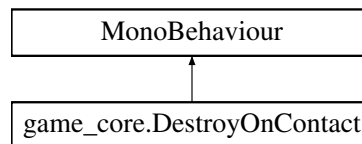
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/CreditsBehaviour.cs

## 5.10 game\_core.DestroyOnContact Class Reference

[DestroyOnContact](#) class defines the behaviour of gameObject when collide with other.

Inheritance diagram for game\_core.DestroyOnContact:



### Public Attributes

- bool **destroy** = true

### 5.10.1 Detailed Description

[DestroyOnContact](#) class defines the behaviour of gameObject when collide with other.

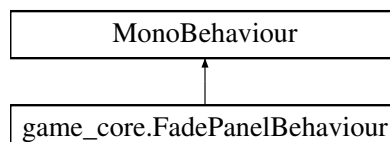
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/DestroyOnContact.cs

## 5.11 game\_core.FadePanelBehaviour Class Reference

Fade panel behaviour class; makes that panels (dis)appear on the screen smoothly.

Inheritance diagram for game\_core.FadePanelBehaviour:



### Public Member Functions

- void [fadeIn](#) ()  
*Fade IN.*

## Public Attributes

- float **fadeTime** = 1.0f
- bool **fadeInFlag** = true

### 5.11.1 Detailed Description

Fade panel behaviour class; makes that panels (dis)appear on the screen smoothly.

### 5.11.2 Member Function Documentation

#### 5.11.2.1 void game\_core.FadePanelBehaviour.fadeIn ( )

Fade IN.

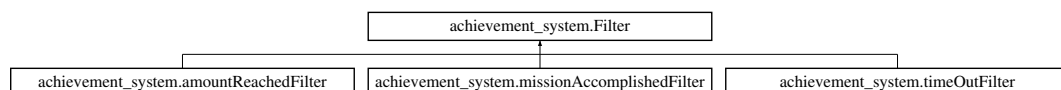
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/FadePanelBehaviour.cs

## 5.12 achievement\_system.Filter Class Reference

[Filter](#) abstract class define a set of operations which all objects that implement class must support.

Inheritance diagram for achievement\_system.Filter:



## Public Member Functions

- [Filter](#) ()  
*Initializes a new instance of the [Filter](#) class.*
- abstract bool [test](#) ()  
*Test this instance.*

### 5.12.1 Detailed Description

[Filter](#) abstract class define a set of operations which all objects that implement class must support.

### 5.12.2 Constructor & Destructor Documentation

#### 5.12.2.1 achievement\_system.Filter.Filter ( )

Initializes a new instance of the [Filter](#) class.



### 5.12.3 Member Function Documentation

#### 5.12.3.1 abstract bool achievement\_system.Filter.test ( ) [pure virtual]

Test this instance.

Implemented in [achievement\\_system.amountReachedFilter](#), [achievement\\_system.missionAccomplishedFilter](#), and [achievement\\_system.timeOutFilter](#).

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/achievement↔ System/Filter.cs

## 5.13 game\_core.FSM Sys< T, E > Class Template Reference

FSMSys class represents the Finite [State](#) Machine class. It has a List with the States the NPC has and methods to add, delete a state, and to change the current state the Machine is on.

### Public Member Functions

- void [AddState](#) ([State](#)< T, E > s)  
*This method places new states inside the FSM, or prints an ERROR message if the state was already inside the List. First state added is also the initial state.*
- void [DeleteState](#) (E id)  
*This method delete a state from the FSM List if it exists, or prints an ERROR message if the state was not on the List.*
- void [PerformTransition](#) (T trans)  
*This method tries to change the state the FSM is in based on the current state and the transition passed. If current state doesn't have a target state for the transition passed, an ERROR message is printed.*

### Public Attributes

- E **nullStateID**
- T **nullTransition**

### Properties

- E **CurrentStateID** [get]
- [State](#)< T, E > **CurrentState** [get]

#### 5.13.1 Detailed Description

FSMSys class represents the Finite [State](#) Machine class. It has a List with the States the NPC has and methods to add, delete a state, and to change the current state the Machine is on.

#### Type Constraints

- T : struct**
- T : IConvertible**
- E : struct**
- E : IConvertible**

### 5.13.2 Member Function Documentation

#### 5.13.2.1 void game\_core.FSMSys< T, E >.AddState ( State< T, E > s )

This method places new states inside the FSM, or prints an ERROR message if the state was already inside the List. First state added is also the initial state.

#### 5.13.2.2 void game\_core.FSMSys< T, E >.DeleteState ( E id )

This method delete a state from the FSM List if it exists, or prints an ERROR message if the state was not on the List.

#### 5.13.2.3 void game\_core.FSMSys< T, E >.PerformTransition ( T trans )

This method tries to change the state the FSM is in based on the current state and the transition passed. If current state doesn't have a target state for the transition passed, an ERROR message is printed.

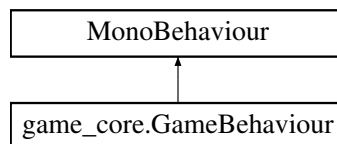
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/plugins/FSMSystem.cs

## 5.14 game\_core.GameBehaviour Class Reference

Game behaviour class is a test system to verify the right operation of code.

Inheritance diagram for game\_core.GameBehaviour:



### Public Attributes

- [missionAccomplishedFilter](#) **saveThePrince**
- [amountReachedFilter](#) **collectedCoins**
- [timeOutFilter](#) **timeOut**

### 5.14.1 Detailed Description

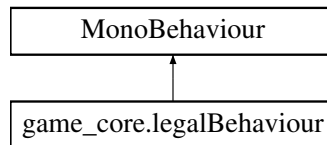
Game behaviour class is a test system to verify the right operation of code.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/Game↔Behaviour.cs

## 5.15 game\_core.legalBehaviour Class Reference

Inheritance diagram for game\_core.legalBehaviour:



### 5.15.1 Detailed Description

Legal behaviour class; manage legal text saved in PlayerPrefs.

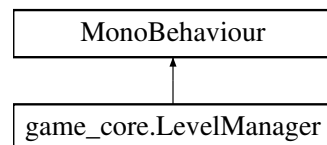
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/legalBehaviour.cs

## 5.16 game\_core.LevelManager Class Reference

Level manager class; Deals with level load transaction.

Inheritance diagram for game\_core.LevelManager:



### Static Public Member Functions

- static void [Load](#) (string name)  
*Load the specified name.*

### Static Protected Attributes

- static [LevelManager](#) **instance**
- static string **levelName** = ""

### Properties

- static [LevelManager Instance](#) [get]  
*Gets the instance.*
- static string [loadingLevel](#) [get, set]  
*Gets or sets the loading level.*

### 5.16.1 Detailed Description

Level manager class; Deals with level load transaction.

## 5.16.2 Member Function Documentation

5.16.2.1 `static void game_core.LevelManager.Load ( string name )` `[static]`

Load the specified name.

## Parameters

<i>name</i>	Name.
-------------	-------

## 5.16.3 Property Documentation

## 5.16.3.1 LevelManager game\_core.LevelManager.Instance [static], [get]

Gets the instance.

The instance.

## 5.16.3.2 string game\_core.LevelManager.loadingLevel [static], [get], [set]

Gets or sets the loading level.

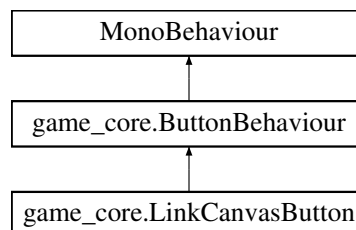
The loading level.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/managers/LevelManager.cs

## 5.17 game\_core.LinkCanvasButton Class Reference

Inheritance diagram for game\_core.LinkCanvasButton:



## Public Attributes

- string **link** = ""

## Protected Member Functions

- override void **action** ()  
*Action this instance.*

## Additional Inherited Members

## 5.17.1 Detailed Description

Link canvas button class; manage external links.

## 5.17.2 Member Function Documentation

### 5.17.2.1 override void game\_core.LinkCanvasButton.action ( ) [protected],[virtual]

Action this instance.

Reimplemented from [game\\_core.ButtonBehaviour](#).

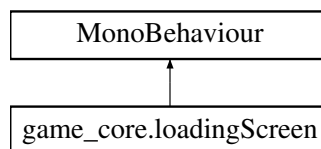
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/LinkCanvasButton.cs

## 5.18 game\_core.loadingScreen Class Reference

Loading screen.

Inheritance diagram for game\_core.loadingScreen:



### Public Attributes

- Color [backgroundColor](#) = Color.black  
*The color of the background.*
- Color [textColor](#) = Color.white  
*The color of the text.*
- string [message](#) = "Loading ..."  
*The message shown on screen.*
- GUIStyle [style](#)  
*The style of GUI.*

### 5.18.1 Detailed Description

Loading screen.

### 5.18.2 Member Data Documentation

#### 5.18.2.1 Color game\_core.loadingScreen.backgroundColor = Color.black

The color of the background.

#### 5.18.2.2 string game\_core.loadingScreen.message = "Loading ..."

The message shown on screen.

#### 5.18.2.3 GUIStyle game\_core.loadingScreen.style

The style of GUI.

#### 5.18.2.4 Color game\_core.loadingScreen.textColor = Color.white

The color of the text.

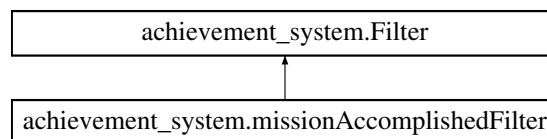
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/loadingScreen.cs

## 5.19 achievement\_system.missionAccomplishedFilter Class Reference

Dichotomic filter; it tests whether the mission is accomplished.

Inheritance diagram for achievement\_system.missionAccomplishedFilter:



### Public Member Functions

- [missionAccomplishedFilter](#) ()  
*Initializes a new instance of the game\_core.timeOutFilter class.*
- override bool [test](#) ()  
*Test this instance.*

### Public Attributes

- bool **missionFlag** = false

### 5.19.1 Detailed Description

Dichotomic filter; it tests whether the mission is accomplished.

### 5.19.2 Constructor & Destructor Documentation

#### 5.19.2.1 achievement\_system.missionAccomplishedFilter.missionAccomplishedFilter ( )

Initializes a new instance of the game\_core.timeOutFilter class.

### 5.19.3 Member Function Documentation

#### 5.19.3.1 override bool achievement\_system.missionAccomplishedFilter.test ( ) [virtual]

Test this instance.

Implements [achievement\\_system.Filter](#).

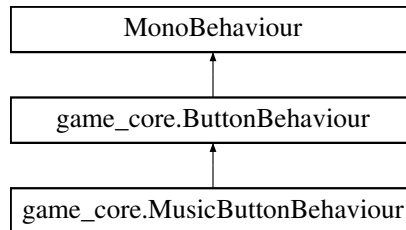
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/achievementSystem/filters.cs

## 5.20 game\_core.MusicButtonBehaviour Class Reference

Music button behaviour class; Enable/Disable music.

Inheritance diagram for game\_core.MusicButtonBehaviour:



### Public Attributes

- Sprite **buttonNormal**
- Sprite **buttonPushed**

### Protected Member Functions

- override void **Start** ()  
*Use this for initialization.*
- override void **action** ()  
*Execute action(OnClick).*

### Additional Inherited Members

#### 5.20.1 Detailed Description

Music button behaviour class; Enable/Disable music.

#### 5.20.2 Member Function Documentation

##### 5.20.2.1 override void game\_core.MusicButtonBehaviour.action ( ) [protected],[virtual]

Execute action(OnClick).

Reimplemented from [game\\_core.ButtonBehaviour](#).

##### 5.20.2.2 override void game\_core.MusicButtonBehaviour.Start ( ) [protected],[virtual]

Use this for initialization.

Reimplemented from [game\\_core.ButtonBehaviour](#).

The documentation for this class was generated from the following file:

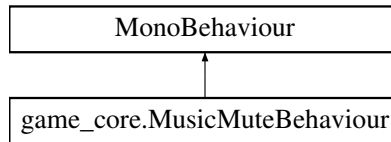
- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/Music↔  
ButtonBehaviour.cs



## 5.21 game\_core.MusicMuteBehaviour Class Reference

Music mute configuration.

Inheritance diagram for game\_core.MusicMuteBehaviour:



### Public Member Functions

- void [setActive](#) (bool value)  
*Enable/Disable audio.*

#### 5.21.1 Detailed Description

Music mute configuration.

#### 5.21.2 Member Function Documentation

##### 5.21.2.1 void game\_core.MusicMuteBehaviour.setActive ( bool value )

Enable/Disable audio.

Parameters

<i>value</i>	If set to <code>true</code> value.
--------------	------------------------------------

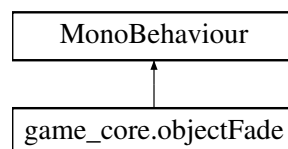
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/MusicMuteBehaviour.cs

## 5.22 game\_core.objectFade Class Reference

Object fade behaviour.

Inheritance diagram for game\_core.objectFade:



### Public Attributes

- Vector3 **startMarker**
- Vector3 **endMarker**
- float **speed** = 1.0F

### 5.22.1 Detailed Description

Object fade behaviour.

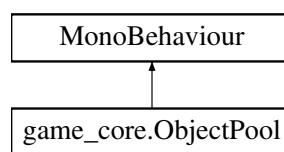
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/object↔ Fade.cs

## 5.23 game\_core.ObjectPool Class Reference

Object pool class creates and manages object instances.

Inheritance diagram for game\_core.ObjectPool:



### Public Member Functions

- GameObject [activateObject](#) ()  
*This method gets the first available gameObject in the array.*

### Public Attributes

- GameObject[] **objectPool**
- int **numberOfObjects** = 0
- GameObject **prefab**

### 5.23.1 Detailed Description

Object pool class creates and manages object instances.

### 5.23.2 Member Function Documentation

#### 5.23.2.1 GameObject game\_core.ObjectPool.activateObject ( )

This method gets the first available gameObject in the array.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/Object↔ Pool.cs

## 5.24 game\_core.PanelView Class Reference

Panel view class; Abstraction layer to deal with UI elements(Panel in this case).

## Public Member Functions

- [PanelView](#) (string name)  
*Initializes a new instance of the [PanelView](#) class.*

## Properties

- GameObject [gameObject](#) [get]  
*Games the object.*
- Transform [transform](#) [get]  
*Transform this instance.*
- RectTransform [rectTransform](#) [get]  
*Transform this instance.*
- Vector3 [position](#) [get]  
*Sets the position.*
- bool [active](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*

### 5.24.1 Detailed Description

Panel view class; Abstraction layer to deal with UI elements(Panel in this case).

### 5.24.2 Constructor & Destructor Documentation

#### 5.24.2.1 game\_core.PanelView.PanelView ( string name )

Initializes a new instance of the [PanelView](#) class.

### 5.24.3 Property Documentation

#### 5.24.3.1 bool game\_core.PanelView.active [get], [set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.

#### 5.24.3.2 GameObject game\_core.PanelView.gameObject [get]

Games the object.

#### Returns

The object.

#### 5.24.3.3 Vector3 game\_core.PanelView.position [get]

Sets the position.

The position.

#### 5.24.3.4 RectTransform game\_core.PanelView.rectTransform [get]

Transform this instance.

#### 5.24.3.5 Transform game\_core.PanelView.transform [get]

Transform this instance.

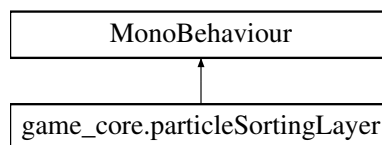
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/views/ui/PanelView.cs

## 5.25 game\_core.particleSortingLayer Class Reference

This class sets the particle sorting layer in 2D perspective.

Inheritance diagram for game\_core.particleSortingLayer:



### Public Attributes

- int **order** = 0
- string **layerName** = "foreground"

#### 5.25.1 Detailed Description

This class sets the particle sorting layer in 2D perspective.

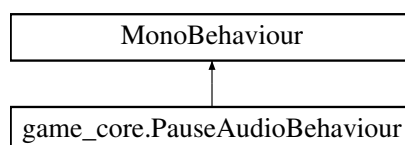
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/particleSortingLayer.cs

## 5.26 game\_core.PauseAudioBehaviour Class Reference

Music behaviour on game pause.

Inheritance diagram for game\_core.PauseAudioBehaviour:



### 5.26.1 Detailed Description

Music behaviour on game pause.

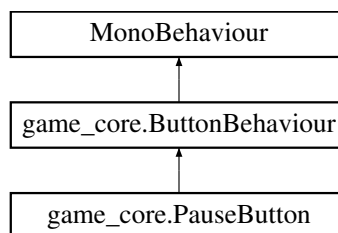
The documentation for this class was generated from the following file:

- `/Users/mrferrys/Documents/looneybits/works/unity/game_core/Assets/game_core/scripts/behaviour/game/PauseAudioBehaviour.cs`

## 5.27 game\_core.PauseButton Class Reference

Pause button.

Inheritance diagram for game\_core.PauseButton:



### Protected Member Functions

- override void `action` ()  
*Execute aciton(OnClick).*

### Additional Inherited Members

### 5.27.1 Detailed Description

Pause button.

### 5.27.2 Member Function Documentation

#### 5.27.2.1 override void game\_core.PauseButton.action ( ) [protected],[virtual]

Execute aciton(OnClick).

Reimplemented from [game\\_core.ButtonBehaviour](#).

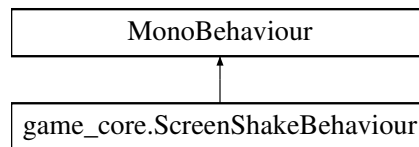
The documentation for this class was generated from the following file:

- `/Users/mrferrys/Documents/looneybits/works/unity/game_core/Assets/game_core/scripts/behaviour/ui/PauseButton.cs`

## 5.28 game\_core.ScreenShakeBehaviour Class Reference

Screen shake effect behaviour.

Inheritance diagram for game\_core.ScreenShakeBehaviour:



## Public Member Functions

- void [OnShake](#) ()  
*Raises the shake event.*

## Public Attributes

- Transform [camTransform](#)  
*Transform of the camera to shake. Grabs the gameObject's transform if null.*
- bool [onEnableShake](#) =true  
*How long the object should shake for.*
- float **shake** = 0f
- float [shakeAmount](#) = 0.7f  
*Amplitude of the shake. A larger value shakes the camera harder.*
- float **decreaseFactor** = 1.0f

### 5.28.1 Detailed Description

Screen shake effect behaviour.

### 5.28.2 Member Function Documentation

#### 5.28.2.1 void game\_core.ScreenShakeBehaviour.OnShake ( )

Raises the shake event.

### 5.28.3 Member Data Documentation

#### 5.28.3.1 Transform game\_core.ScreenShakeBehaviour.camTransform

Transform of the camera to shake. Grabs the gameObject's transform if null.

#### 5.28.3.2 bool game\_core.ScreenShakeBehaviour.onEnableShake =true

How long the object should shake for.

#### 5.28.3.3 float game\_core.ScreenShakeBehaviour.shakeAmount = 0.7f

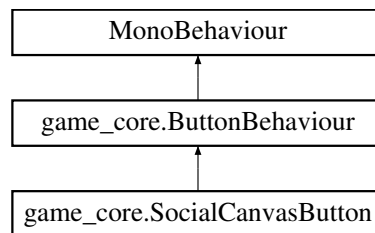
Amplitude of the shake. A larger value shakes the camera harder.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/game/ScreenShakeBehaviour.cs

## 5.29 game\_core.SocialCanvasButton Class Reference

Inheritance diagram for game\_core.SocialCanvasButton:



### Public Attributes

- string **title** = "GAME TITLE"
- string **url** = "http://looneybits.com"
- string **caption** = "GAME TITLE"
- string **description** = "Super Score: "
- string **hashtags** = "#looneybits"
- string **variableName** = "superScore"
- string **FBAppID** = "1435061540094302"
- string **FBRedirectUri** = "http://looneybits.com"

### Protected Member Functions

- override void **Start** ()  
*Use this for initialization.*
- override void **action** ()  
*PROBLEM: Right now the best solution to solve the problem of share button with facebook is use dialog system. Sharer system has suffered some changes.*

### Additional Inherited Members

#### 5.29.1 Detailed Description

Social canvas button class; Share in social media the player score.

#### 5.29.2 Member Function Documentation

##### 5.29.2.1 override void game\_core.SocialCanvasButton.action ( ) [protected],[virtual]

PROBLEM: Right now the best solution to solve the problem of share button with facebook is use dialog system. Sharer system has suffered some changes.

1 SOLUTION SAMPLE OF FACEBOOK DIALOGS:(CHANGE APP\_ID) [http://www.facebook.com/dialog/feed? app\\_id=123050457758183& link=http://developers.facebook.com/docs/reference/dialogs& picture=http://looneybits.com/assets/img/p04.png& name=Facebook%20Dialogs& caption=Reference%20Documentation& description=Dialogs%20provide%20a%20simple,%20consistent%20interface%20for%20applications%20to%20integrate%20with%20facebook& message=Facebook%20Dialogs%20are%20so%20easy!& redirect\\_uri=http://www.example.com/response](http://www.facebook.com/dialog/feed?app_id=123050457758183&link=http://developers.facebook.com/docs/reference/dialogs&picture=http://looneybits.com/assets/img/p04.png&name=Facebook%20Dialogs&caption=Reference%20Documentation&description=Dialogs%20provide%20a%20simple,%20consistent%20interface%20for%20applications%20to%20integrate%20with%20facebook&message=Facebook%20Dialogs%20are%20so%20easy!&redirect_uri=http://www.example.com/response)

REQUISITES(TWITTER DOES NOT HAVE REQUISITES) FACEBOOK ACCOUNT REGISTER AS FACEBOOK DEVELOPER APP\_ID YOU MUST TO HAVE AN APP\_ID. REGISTER A SIMPLE APP ON <http://developers.facebook.com>

2 SOLUTION SAMPLE OF FACEBOOK SHARER.php (DEPRECATED HALF SOLUTION) [http://www.facebook.com/sharer.php?s=100&p\[title\]=TITLE&p\[url\]=http://looneybits.com&p\[summary\]=yoursummaryhere&p\[images\]\[0\]=http://looneybits.com/assets/img/p04.png](http://www.facebook.com/sharer.php?s=100&p[title]=TITLE&p[url]=http://looneybits.com&p[summary]=yoursummaryhere&p[images][0]=http://looneybits.com/assets/img/p04.png);

LITERATURE:

<http://stackoverflow.com/questions/5023602/facebook-share-body-text>

WITH OLD SHARER.php

```
string facebookshare = "https://www.facebook.com/sharer/sharer.php?t="+System.Uri.EscapeDataString(title)+"&u=";
string facebookshare = "http://www.facebook.com/sharer.php?s=100&p[title]=TITLE&p[url]=http://looneybits.com&p[summary]=yoursummaryhere&p[images][0]=http://looneybits.com/assets/img/p04.png";
```

Reimplemented from [game\\_core.ButtonBehaviour](#).

**5.29.2.2** override void `game_core.SocialCanvasButton.Start ( )` [protected], [virtual]

Use this for initialization.

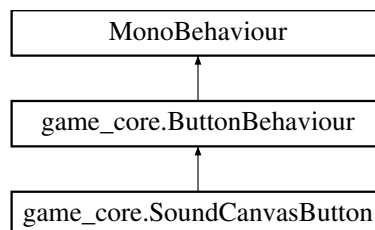
Reimplemented from [game\\_core.ButtonBehaviour](#).

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/SocialCanvasButton.cs

## 5.30 game\_core.SoundCanvasButton Class Reference

Inheritance diagram for `game_core.SoundCanvasButton`:



### Public Attributes

- Sprite **buttonNormal**
- Sprite **buttonPushed**

### Protected Member Functions

- override void [Start](#) ()  
*Use this for initialization.*
- override void [action](#) ()  
*Action this instance.*

### Additional Inherited Members

#### 5.30.1 Detailed Description

Sound canvas button class; Enable/Disable sound.



## 5.30.2 Member Function Documentation

### 5.30.2.1 override void game\_core.SoundCanvasButton.action ( ) [protected],[virtual]

Action this instance.

Reimplemented from [game\\_core.ButtonBehaviour](#).

### 5.30.2.2 override void game\_core.SoundCanvasButton.Start ( ) [protected],[virtual]

Use this for initialization.

Reimplemented from [game\\_core.ButtonBehaviour](#).

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/SoundCanvasButton.cs

## 5.31 game\_core.SoundView Class Reference

Sound view class; Abstraction layer to deal with UI elements(Sound in this case).

### Public Member Functions

- [SoundView](#) (string name)  
*Initializes a new instance of the [SoundView](#) class.*

### Properties

- GameObject [gameObject](#) [get]  
*Games the object.*
- Transform [transform](#) [get]  
*Transform this instance.*
- AudioSource [audioSource](#) [get]  
*Transform this instance.*
- Vector3 [position](#) [get]  
*Sets the position.*
- bool [active](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*
- bool [play](#) [get, set]  
*Gets or sets a value indicating whether this [SoundView](#) is play.*
- float [volume](#) [get, set]  
*Gets or sets the volume.*
- float [pitch](#) [get, set]  
*Gets or sets the pitch.*

### 5.31.1 Detailed Description

Sound view class; Abstraction layer to deal with UI elements(Sound in this case).

### 5.31.2 Constructor & Destructor Documentation

#### 5.31.2.1 `game_core.SoundView.SoundView ( string name )`

Initializes a new instance of the [SoundView](#) class.

### 5.31.3 Property Documentation

#### 5.31.3.1 `bool game_core.SoundView.active [get], [set]`

Gets or sets a value indicating whether this IndicatorView is active.

`true` if active; otherwise, `false`.

#### 5.31.3.2 `AudioSource game_core.SoundView.audioSource [get]`

Transform this instance.

#### 5.31.3.3 `GameObject game_core.SoundView.gameObject [get]`

Games the object.

##### Returns

The object.

#### 5.31.3.4 `float game_core.SoundView.pitch [get], [set]`

Gets or sets the pitch.

The pitch.

#### 5.31.3.5 `bool game_core.SoundView.play [get], [set]`

Gets or sets a value indicating whether this [SoundView](#) is play.

`true` if play; otherwise, `false`.

#### 5.31.3.6 `Vector3 game_core.SoundView.position [get]`

Sets the position.

The position.

#### 5.31.3.7 `Transform game_core.SoundView.transform [get]`

Transform this instance.

#### 5.31.3.8 `float game_core.SoundView.volume [get], [set]`

Gets or sets the volume.

The volume.

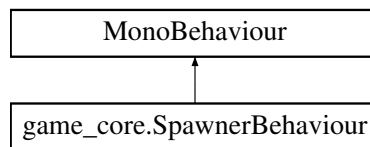
The documentation for this class was generated from the following file:

- [/Users/mrferrys/Documents/looneybits/works/unity/game\\_core/Assets/game\\_core/scripts/views/ui/SoundView.cs](#)

## 5.32 game\_core.SpawnerBehaviour Class Reference

This class spawns objects with a certain force and every X seconds(timeRange).

Inheritance diagram for game\_core.SpawnerBehaviour:



### Public Attributes

- Vector2 **timeRange** = new Vector2(5.0f,10.0f)
- Vector2 **force** = new Vector2(0,-200)

#### 5.32.1 Detailed Description

This class spawns objects with a certain force and every X seconds(timeRange).

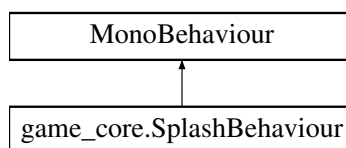
The documentation for this class was generated from the following file:

- [/Users/mrferrys/Documents/looneybits/works/unity/game\\_core/Assets/game\\_core/scripts/behaviour/game/SpawnerBehaviour.cs](#)

## 5.33 game\_core.SplashBehaviour Class Reference

Splash behaviour.

Inheritance diagram for game\_core.SplashBehaviour:



### Public Attributes

- float **timeOut** = 1.0f
- string **levelName** = "menu"

#### 5.33.1 Detailed Description

Splash behaviour.

The documentation for this class was generated from the following file:

- `/Users/mrferrys/Documents/looneybits/works/unity/game_core/Assets/game_core/scripts/behaviour/game/SplashBehaviour.cs`

## 5.34 game\_core.StarView Class Reference

Star view class; Abstraction layer to deal with UI elements(Star in this case).

### Public Member Functions

- [StarView](#) (string name)  
*Initializes a new instance of the [BarView](#) class.*

### Properties

- GameObject [gameObject](#) [get]  
*Games the object.*
- Transform [transform](#) [get]  
*Transform this instance.*
- RectTransform [rectTransform](#) [get]  
*Transform this instance.*
- Vector3 [position](#) [get]  
*Sets the position.*
- bool [active](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*
- bool [activeStar](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*

#### 5.34.1 Detailed Description

Star view class; Abstraction layer to deal with UI elements(Star in this case).

#### 5.34.2 Constructor & Destructor Documentation

##### 5.34.2.1 game\_core.StarView.StarView ( string name )

Initializes a new instance of the [BarView](#) class.

#### 5.34.3 Property Documentation

##### 5.34.3.1 bool game\_core.StarView.active [get], [set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.

##### 5.34.3.2 bool game\_core.StarView.activeStar [get], [set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.

## 5.34.3.3 GameObject game\_core.StarView.gameObject [get]

Games the object.

## Returns

The object.

## 5.34.3.4 Vector3 game\_core.StarView.position [get]

Sets the position.

The position.

## 5.34.3.5 RectTransform game\_core.StarView.rectTransform [get]

Transform this instance.

## 5.34.3.6 Transform game\_core.StarView.transform [get]

Transform this instance.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/views/ui/StarView.cs

## 5.35 game\_core.State&lt; T, E &gt; Class Template Reference

This class represents the States in the Finite [State](#) System. Each state has a Dictionary with pairs (transition-state) showing which state the FSM should be if a transition is fired while this state is the current state. Method Reason is used to determine which transition should be fired . Method Act has the code to perform the actions the NPC is supposed do if it's on this state.

## Public Member Functions

- void **AddTransition** (T trans, E id)
- void **DeleteTransition** (T trans)
 

*This method deletes a pair transition-state from this state's map. If the transition was not inside the state's map, an ERROR message is printed.*
- E **GetOutputState** (T trans)
 

*This method returns the new state the FSM should be if this state receives a transition and*
- virtual void **DoBeforeEntering** ()
 

*This method is used to set up the [State](#) condition before entering it. It is called automatically by the FSMSystem class before assigning it to the current state.*
- virtual void **DoBeforeLeaving** ()
 

*This method is used to make anything necessary, as resetting variables before the FSMSystem changes to another one. It is called automatically by the FSMSystem before changing to a new state.*
- abstract void **Reason** ()
 

*This method decides if the state should transition to another on its list NPC is a reference to the object that is controlled by this class*
- abstract void **Act** ()
 

*This method controls the behavior of the NPC in the game World. Every action, movement or communication the NPC does should be placed here NPC is a reference to the object that is controlled by this class*

## Public Attributes

- **T nullTransition**
- **E nullStateID**

## Protected Attributes

- Dictionary< T, E > **map** = new Dictionary<T, E>()
- **E stateID**

## Properties

- **E ID** [get]

### 5.35.1 Detailed Description

This class represents the States in the Finite [State](#) System. Each state has a Dictionary with pairs (transition-state) showing which state the FSM should be if a transition is fired while this state is the current state. Method Reason is used to determine which transition should be fired . Method Act has the code to perform the actions the NPC is supposed do if it's on this state.

A Finite [State](#) Machine System based on Chapter 3.1 of Game Programming Gems 1 by Eric Dybsand

Written by Roberto Cezar Bianchini, July 2010

How to use:

1. Place the labels for the transitions and the states of the Finite [State](#) System in the corresponding enums.
2. Write new class(es) inheriting from FSMState and fill each one with pairs (transition-state). These pairs represent the state S2 the FSMSystem should be if while being on state S1, a transition T is fired and state S1 has a transition from it to S2. Remember this is a Deterministic FSM. You can't have one transition leading to two different states.  
  
Method Reason is used to determine which transition should be fired. You can write the code to fire transitions in another place, and leave this method empty if you feel it's more appropriate to your project.  
  
Method Act has the code to perform the actions the NPC is supposed do if it's on this state. You can write the code for the actions in another place, and leave this method empty if you feel it's more appropriate to your project.
3. Create an instance of FSMSystem class and add the states to it.
4. Call Reason and Act (or whichever methods you have for firing transitions and making the NPCs behave in your game) from your Update or FixedUpdate methods.

Asynchronous transitions from Unity Engine, like OnTriggerEnter, SendMessage, can also be used, just call the Method PerformTransition from your FSMSystem instance with the correct Transition when the event occurs.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Type Constraints

- T : struct**
- T : IConvertible**
- E : struct**
- E : IConvertible**

### 5.35.2 Member Function Documentation

#### 5.35.2.1 abstract void game\_core.State< T, E >.Act ( ) [pure virtual]

This method controls the behavior of the NPC in the game World. Every action, movement or communication the NPC does should be placed here NPC is a reference to the object that is controlled by this class

#### 5.35.2.2 void game\_core.State< T, E >.DeleteTransition ( T *trans* )

This method deletes a pair transition-state from this state's map. If the transition was not inside the state's map, an ERROR message is printed.

#### 5.35.2.3 virtual void game\_core.State< T, E >.DoBeforeEntering ( ) [virtual]

This method is used to set up the [State](#) condition before entering it. It is called automatically by the FSMSystem class before assigning it to the current state.

#### 5.35.2.4 virtual void game\_core.State< T, E >.DoBeforeLeaving ( ) [virtual]

This method is used to make anything necessary, as resetting variables before the FSMSystem changes to another one. It is called automatically by the FSMSystem before changing to a new state.

#### 5.35.2.5 E game\_core.State< T, E >.GetOutputState ( T *trans* )

This method returns the new state the FSM should be if this state receives a transition and

#### 5.35.2.6 abstract void game\_core.State< T, E >.Reason ( ) [pure virtual]

This method decides if the state should transition to another on its list NPC is a reference to the object that is controlled by this class

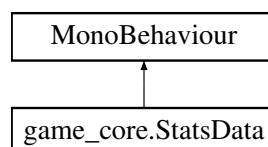
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/plugins/FSMSystem.cs

## 5.36 game\_core.StatsData Class Reference

Stats data class; manages the stats saved in PlayerPrefs.

Inheritance diagram for game\_core.StatsData:



### Public Attributes

- string **variableName** = ""

### 5.36.1 Detailed Description

Stats data class; manages the stats saved in PlayerPrefs.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/StatsData.cs

## 5.37 game\_core.TextView Class Reference

Text view class; Abstraction layer to deal with UI elements(Text in this case).

### Public Member Functions

- [TextView](#) (string name)  
*Initializes a new instance of the [TextView](#) class.*

### Properties

- GameObject [gameObject](#) [get]  
*Games the object.*
- Transform [transform](#) [get]  
*Transform this instance.*
- RectTransform [rectTransform](#) [get]  
*Transform this instance.*
- Vector3 [position](#) [get]  
*Sets the position.*
- bool [active](#) [get, set]  
*Gets or sets a value indicating whether this IndicatorView is active.*
- string [text](#) [set]  
*Sets the text.*

### 5.37.1 Detailed Description

Text view class; Abstraction layer to deal with UI elements(Text in this case).

### 5.37.2 Constructor & Destructor Documentation

#### 5.37.2.1 game\_core.TextView.TextView ( string name )

Initializes a new instance of the [TextView](#) class.

### 5.37.3 Property Documentation

#### 5.37.3.1 bool game\_core.TextView.active [get], [set]

Gets or sets a value indicating whether this IndicatorView is active.

true if active; otherwise, false.



## 5.37.3.2 GameObject game\_core.TextView.gameObject [get]

Games the object.

Returns

The object.

## 5.37.3.3 Vector3 game\_core.TextView.position [get]

Sets the position.

The position.

## 5.37.3.4 RectTransform game\_core.TextView.rectTransform [get]

Transform this instance.

## 5.37.3.5 string game\_core.TextView.text [set]

Sets the text.

The text.

## 5.37.3.6 Transform game\_core.TextView.transform [get]

Transform this instance.

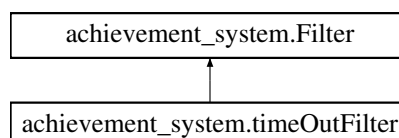
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/views/ui/Text↔View.cs

## 5.38 achievement\_system.timeOutFilter Class Reference

time filter; it tests whether the time is over.

Inheritance diagram for achievement\_system.timeOutFilter:



## Public Member Functions

- [timeOutFilter](#) ()  
*Initializes a new instance of the game\_core.timeOutFilter class.*
- override bool [test](#) ()  
*Test this instance.*

## Public Attributes

- float **timeLimit** = 0.0f

## Properties

- float **time** [get, set]  
*Gets or sets the time.*

### 5.38.1 Detailed Description

time filter; it tests whether the time is over.

### 5.38.2 Constructor & Destructor Documentation

#### 5.38.2.1 achievement\_system.timeOutFilter.timeOutFilter ( )

Initializes a new instance of the game\_core.timeOutFilter class.

### 5.38.3 Member Function Documentation

#### 5.38.3.1 override bool achievement\_system.timeOutFilter.test ( ) [virtual]

Test this instance.

Implements [achievement\\_system.Filter](#).

### 5.38.4 Property Documentation

#### 5.38.4.1 float achievement\_system.timeOutFilter.time [get], [set]

Gets or sets the time.

The time.

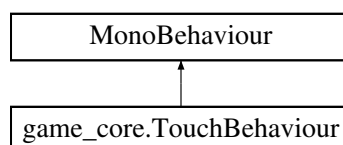
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/achievement↔  
System/filters.cs

## 5.39 game\_core.TouchBehaviour Class Reference

Touch behaviour class; Not in use, but define a set of operations which all objects that implement class must support (similar [ButtonBehaviour](#) class).

Inheritance diagram for game\_core.TouchBehaviour:



## Protected Member Functions

- virtual void [Start](#) ()  
*Use this for initialization*
- virtual void [Update](#) ()  
*Update is called once per frame*
- virtual void [OnTouchBegan](#) (Vector3 v)  
*Raises the touch began event.*
- virtual void [OnTouchCanceled](#) (Vector3 v)  
*Raises the touch canceled event.*
- virtual void [OnTouchEnded](#) (Vector3 v)  
*Raises the touch ended event.*
- virtual void [OnTouchMoved](#) (Vector3 v)  
*Raises the touch moved event.*
- virtual void [OnTouchStay](#) (Vector3 v)  
*Raises the touch stay event.*
- virtual void **action** ()

### 5.39.1 Detailed Description

Touch behaviour class; Not in use, but define a set of operations which all objects that implement class must support (similar [ButtonBehaviour](#) class).

### 5.39.2 Member Function Documentation

5.39.2.1 virtual void game\_core.TouchBehaviour.OnTouchBegan ( Vector3 v ) [protected],[virtual]

Raises the touch began event.

Parameters

v	V.
---	----

5.39.2.2 virtual void game\_core.TouchBehaviour.OnTouchCanceled ( Vector3 v ) [protected],[virtual]

Raises the touch canceled event.

Parameters

v	V.
---	----

5.39.2.3 virtual void game\_core.TouchBehaviour.OnTouchEnded ( Vector3 v ) [protected],[virtual]

Raises the touch ended event.

Parameters

v	V.
---	----

5.39.2.4 virtual void game\_core.TouchBehaviour.OnTouchMoved ( Vector3 v ) [protected],[virtual]

Raises the touch moved event.

## Parameters

$v$	$V.$
-----	------

5.39.2.5 `virtual void game_core.TouchBehaviour.OnTouchStay ( Vector3  $v$  )` [protected],[virtual]

Raises the touch stay event.

## Parameters

$v$	$V.$
-----	------

5.39.2.6 `virtual void game_core.TouchBehaviour.Start ( )` [protected],[virtual]

Use this for initialization

5.39.2.7 `virtual void game_core.TouchBehaviour.Update ( )` [protected],[virtual]

Update is called once per frame

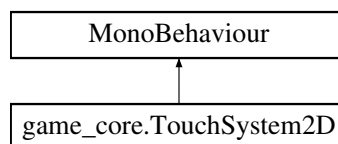
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/behaviour/ui/TouchBehaviour.cs

## 5.40 game\_core.TouchSystem2D Class Reference

Touch system2d(Vector2) class; Deals with the hits on the screen.

Inheritance diagram for game\_core.TouchSystem2D:



### Public Attributes

- LayerMask `touchInputMask`  
*Layer mask name for example input.*
- Camera **camera**

### 5.40.1 Detailed Description

Touch system2d(Vector2) class; Deals with the hits on the screen.

### 5.40.2 Member Data Documentation

5.40.2.1 LayerMask `game_core.TouchSystem2D.touchInputMask`

Layer mask name for example input.

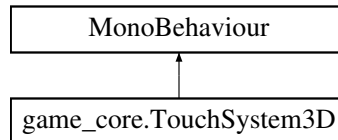
The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/touchSystem/TouchSystem2D.cs

## 5.41 game\_core.TouchSystem3D Class Reference

Touch system3d(Vector3) class; Deals with the hits on the screen.

Inheritance diagram for game\_core.TouchSystem3D:



### Public Attributes

- LayerMask [touchInputMask](#)  
*Layer mask name for example input.*
- Camera **camera**

#### 5.41.1 Detailed Description

Touch system3d(Vector3) class; Deals with the hits on the screen.

#### 5.41.2 Member Data Documentation

##### 5.41.2.1 LayerMask game\_core.TouchSystem3D.touchInputMask

Layer mask name for example input.

The documentation for this class was generated from the following file:

- /Users/mrferrys/Documents/looneybits/works/unity/game\_core/Assets/game\_core/scripts/touchSystem/TouchSystem3D.cs



# Index

- achievement\_system, 7
- achievement\_system.achievement, 11
- achievement\_system.actionBehaviour, 12
- achievement\_system.amountReachedFilter, 12
- achievement\_system.Filter, 22
- achievement\_system.missionAccomplishedFilter, 29
- achievement\_system.timeOutFilter, 47
- achievement\_system::Filter
  - Filter, 22
  - test, 23
- achievement\_system::achievement
  - Awake, 12
  - filter, 12
  - isAccomplished, 12
  - Start, 12
- achievement\_system::amountReachedFilter
  - amountReachedFilter, 13
  - test, 13
  - value, 13
- achievement\_system::missionAccomplishedFilter
  - missionAccomplishedFilter, 29
  - test, 29
- achievement\_system::timeOutFilter
  - test, 48
  - time, 48
  - timeOutFilter, 48
- Act
  - game\_core::State, 45
- action
  - game\_core::CanvasButton, 20
  - game\_core::LinkCanvasButton, 28
  - game\_core::MusicButtonBehaviour, 30
  - game\_core::PauseButton, 35
  - game\_core::SocialCanvasButton, 37
  - game\_core::SoundCanvasButton, 39
- activateObject
  - game\_core::ObjectPool, 32
- active
  - game\_core::BarView, 14
  - game\_core::ButtonView, 18
  - game\_core::PanelView, 33
  - game\_core::SoundView, 40
  - game\_core::StarView, 42
  - game\_core::TextView, 46
- activeStar
  - game\_core::StarView, 42
- AddState
  - game\_core::FSMSys, 24
- amountReachedFilter
  - achievement\_system::amountReachedFilter, 13
- audioSource
  - game\_core::SoundView, 40
- Awake
  - achievement\_system::achievement, 12
- backgroundColor
  - game\_core::loadingScreen, 28
- BarView
  - game\_core::BarView, 14
- ButtonView
  - game\_core::ButtonView, 18
- camTransform
  - game\_core::ScreenShakeBehaviour, 36
- DeleteState
  - game\_core::FSMSys, 24
- DeleteTransition
  - game\_core::State, 45
- DoBeforeEntering
  - game\_core::State, 45
- DoBeforeLeaving
  - game\_core::State, 45
- fadeOut
  - game\_core::FadePanelBehaviour, 22
- Filter
  - achievement\_system::Filter, 22
- filter
  - achievement\_system::achievement, 12
- FixedUpdate
  - game\_core::ButtonBehaviour, 17
- game\_core, 7
- game\_core.BarView, 14
- game\_core.BoundaryBehaviour, 15
- game\_core.ButtonBehaviour, 16
- game\_core.ButtonView, 18
- game\_core.CanvasButton, 19
- game\_core.CreditsBehaviour, 20
- game\_core.DestroyOnContact, 21
- game\_core.FSMSys< T, E >, 23
- game\_core.FadePanelBehaviour, 21
- game\_core.GameBehaviour, 24
- game\_core.legalBehaviour, 24
- game\_core.LevelManager, 25
- game\_core.LinkCanvasButton, 27
- game\_core.loadingScreen, 28
- game\_core.MusicButtonBehaviour, 30
- game\_core.MusicMuteBehaviour, 31

- game\_core.objectFade, [31](#)
- game\_core.ObjectPool, [32](#)
- game\_core.PanelView, [32](#)
- game\_core.particleSortingLayer, [34](#)
- game\_core.PauseAudioBehaviour, [34](#)
- game\_core.PauseButton, [35](#)
- game\_core.ScreenShakeBehaviour, [35](#)
- game\_core.SocialCanvasButton, [37](#)
- game\_core.SoundCanvasButton, [38](#)
- game\_core.SoundView, [39](#)
- game\_core.SpawnerBehaviour, [41](#)
- game\_core.SplashBehaviour, [41](#)
- game\_core.StarView, [42](#)
- game\_core.State< T, E >, [43](#)
- game\_core.StatsData, [45](#)
- game\_core.TextView, [46](#)
- game\_core.TouchBehaviour, [48](#)
- game\_core.TouchSystem2D, [50](#)
- game\_core.TouchSystem3D, [51](#)
- game\_core::BarView
  - active, [14](#)
  - BarView, [14](#)
  - gameObject, [14](#)
  - position, [15](#)
  - rectTransform, [15](#)
  - text, [15](#)
  - transform, [15](#)
  - value, [15](#)
- game\_core::ButtonBehaviour
  - FixedUpdate, [17](#)
  - OnClickEvent, [17](#)
  - OnEnable, [17](#)
  - OnMouseDown, [17](#)
  - OnMouseDrag, [17](#)
  - OnMouseEnter, [17](#)
  - OnMouseExit, [17](#)
  - OnMouseOver, [17](#)
  - OnMouseUp, [17](#)
  - Start, [17](#)
  - Update, [17](#)
- game\_core::ButtonView
  - active, [18](#)
  - ButtonView, [18](#)
  - gameObject, [18](#)
  - interactable, [19](#)
  - position, [19](#)
  - rectTransform, [19](#)
  - transform, [19](#)
- game\_core::CanvasButton
  - action, [20](#)
- game\_core::CreditsBehaviour
  - OnClick, [21](#)
- game\_core::FSMSys
  - AddState, [24](#)
  - DeleteState, [24](#)
  - PerformTransition, [24](#)
- game\_core::FadePanelBehaviour
  - fadeOut, [22](#)
- game\_core::LevelManager
  - Instance, [27](#)
  - Load, [26](#)
  - loadingLevel, [27](#)
- game\_core::LinkCanvasButton
  - action, [28](#)
- game\_core::MusicButtonBehaviour
  - action, [30](#)
  - Start, [30](#)
- game\_core::MusicMuteBehaviour
  - setActive, [31](#)
- game\_core::ObjectPool
  - activateObject, [32](#)
- game\_core::PanelView
  - active, [33](#)
  - gameObject, [33](#)
  - PanelView, [33](#)
  - position, [33](#)
  - rectTransform, [33](#)
  - transform, [34](#)
- game\_core::PauseButton
  - action, [35](#)
- game\_core::ScreenShakeBehaviour
  - camTransform, [36](#)
  - onEnableShake, [36](#)
  - OnShake, [36](#)
  - shakeAmount, [36](#)
- game\_core::SocialCanvasButton
  - action, [37](#)
  - Start, [38](#)
- game\_core::SoundCanvasButton
  - action, [39](#)
  - Start, [39](#)
- game\_core::SoundView
  - active, [40](#)
  - audioSource, [40](#)
  - gameObject, [40](#)
  - pitch, [40](#)
  - play, [40](#)
  - position, [40](#)
  - SoundView, [40](#)
  - transform, [40](#)
  - volume, [40](#)
- game\_core::StarView
  - active, [42](#)
  - activeStar, [42](#)
  - gameObject, [42](#)
  - position, [43](#)
  - rectTransform, [43](#)
  - StarView, [42](#)
  - transform, [43](#)
- game\_core::State
  - Act, [45](#)
  - DeleteTransition, [45](#)
  - DoBeforeEntering, [45](#)
  - DoBeforeLeaving, [45](#)
  - GetOutputState, [45](#)
  - Reason, [45](#)



- game\_core::TextView
  - active, 46
  - gameObject, 46
  - position, 47
  - rectTransform, 47
  - text, 47
  - TextView, 46
  - transform, 47
- game\_core::TouchBehaviour
  - OnTouchBegan, 49
  - OnTouchCanceled, 49
  - OnTouchEnded, 49
  - OnTouchMoved, 49
  - OnTouchStay, 50
  - Start, 50
  - Update, 50
- game\_core::TouchSystem2D
  - touchInputMask, 50
- game\_core::TouchSystem3D
  - touchInputMask, 51
- game\_core::loadingScreen
  - backgroundColor, 28
  - message, 28
  - style, 28
  - textColor, 28
- gameObject
  - game\_core::BarView, 14
  - game\_core::ButtonView, 18
  - game\_core::PanelView, 33
  - game\_core::SoundView, 40
  - game\_core::StarView, 42
  - game\_core::TextView, 46
- GetOutputState
  - game\_core::State, 45
- Instance
  - game\_core::LevelManager, 27
- interactable
  - game\_core::ButtonView, 19
- isAccomplished
  - achievement\_system::achievement, 12
- Load
  - game\_core::LevelManager, 26
- loadingLevel
  - game\_core::LevelManager, 27
- message
  - game\_core::loadingScreen, 28
- missionAccomplishedFilter
  - achievement\_system::missionAccomplishedFilter, 29
- OnClick
  - game\_core::CreditsBehaviour, 21
- OnClickEvent
  - game\_core::ButtonBehaviour, 17
- OnEnable
  - game\_core::ButtonBehaviour, 17
- onEnableShake
  - game\_core::ScreenShakeBehaviour, 36
- OnMouseDown
  - game\_core::ButtonBehaviour, 17
- OnMouseDownDrag
  - game\_core::ButtonBehaviour, 17
- OnMouseEnter
  - game\_core::ButtonBehaviour, 17
- OnMouseExit
  - game\_core::ButtonBehaviour, 17
- OnMouseOver
  - game\_core::ButtonBehaviour, 17
- OnMouseUp
  - game\_core::ButtonBehaviour, 17
- OnShake
  - game\_core::ScreenShakeBehaviour, 36
- OnTouchBegan
  - game\_core::TouchBehaviour, 49
- OnTouchCanceled
  - game\_core::TouchBehaviour, 49
- OnTouchEnded
  - game\_core::TouchBehaviour, 49
- OnTouchMoved
  - game\_core::TouchBehaviour, 49
- OnTouchStay
  - game\_core::TouchBehaviour, 50
- PanelView
  - game\_core::PanelView, 33
- PerformTransition
  - game\_core::FSMSys, 24
- pitch
  - game\_core::SoundView, 40
- play
  - game\_core::SoundView, 40
- position
  - game\_core::BarView, 15
  - game\_core::ButtonView, 19
  - game\_core::PanelView, 33
  - game\_core::SoundView, 40
  - game\_core::StarView, 43
  - game\_core::TextView, 47
- Reason
  - game\_core::State, 45
- rectTransform
  - game\_core::BarView, 15
  - game\_core::ButtonView, 19
  - game\_core::PanelView, 33
  - game\_core::StarView, 43
  - game\_core::TextView, 47
- setActive
  - game\_core::MusicMuteBehaviour, 31
- shakeAmount
  - game\_core::ScreenShakeBehaviour, 36
- SoundView
  - game\_core::SoundView, 40
- StarView

- game\_core::StarView, [42](#)
- Start
  - achievement\_system::achievement, [12](#)
  - game\_core::ButtonBehaviour, [17](#)
  - game\_core::MusicButtonBehaviour, [30](#)
  - game\_core::SocialCanvasButton, [38](#)
  - game\_core::SoundCanvasButton, [39](#)
  - game\_core::TouchBehaviour, [50](#)
- style
  - game\_core::loadingScreen, [28](#)
- test
  - achievement\_system::Filter, [23](#)
  - achievement\_system::amountReachedFilter, [13](#)
  - achievement\_system::missionAccomplishedFilter, [29](#)
  - achievement\_system::timeOutFilter, [48](#)
- text
  - game\_core::BarView, [15](#)
  - game\_core::TextView, [47](#)
- textColor
  - game\_core::loadingScreen, [28](#)
- TextView
  - game\_core::TextView, [46](#)
- time
  - achievement\_system::timeOutFilter, [48](#)
- timeOutFilter
  - achievement\_system::timeOutFilter, [48](#)
- touchInputMask
  - game\_core::TouchSystem2D, [50](#)
  - game\_core::TouchSystem3D, [51](#)
- transform
  - game\_core::BarView, [15](#)
  - game\_core::ButtonView, [19](#)
  - game\_core::PanelView, [34](#)
  - game\_core::SoundView, [40](#)
  - game\_core::StarView, [43](#)
  - game\_core::TextView, [47](#)
- Update
  - game\_core::ButtonBehaviour, [17](#)
  - game\_core::TouchBehaviour, [50](#)
- value
  - achievement\_system::amountReachedFilter, [13](#)
  - game\_core::BarView, [15](#)
- volume
  - game\_core::SoundView, [40](#)