Anibody

A 2d Canvas-based game engine

*Version: 1.1.7*

Inhaltsverzeichnis

[Preface 4](#_Toc503386168)

[Who wants to use Anibody? 4](#_Toc503386169)

[Idea of the Engine 4](#_Toc503386170)

[Installation 4](#_Toc503386171)

[Basic Concept 4](#_Toc503386172)

[The Game Cycle - here Frame() 5](#_Toc503386173)

[Documentation 5](#_Toc503386174)

[Anibody 6](#_Toc503386175)

[Anibody.\* 6](#_Toc503386176)

[EngineObject 6](#_Toc503386177)

[ABO 6](#_Toc503386178)

[Widget 6](#_Toc503386179)

[DefaultCamera 6](#_Toc503386180)

[DefaultTerrain 6](#_Toc503386181)

[ECMAScriptExtension 6](#_Toc503386182)

[Anibody.debug.\* 6](#_Toc503386183)

[DumbObject 6](#_Toc503386184)

[Consolero 6](#_Toc503386185)

[DebugWindow 6](#_Toc503386186)

[Monitor 6](#_Toc503386187)

[Anibody.input.\* 6](#_Toc503386188)

[Anibody.nav.\* 7](#_Toc503386189)

[BoxMenu 7](#_Toc503386190)

[Gallery 7](#_Toc503386191)

[SlideMenu > Tab 7](#_Toc503386192)

[Anibody.shapes.\* 7](#_Toc503386193)

[Shape 7](#_Toc503386194)

[Rectangle 7](#_Toc503386195)

[Circle 7](#_Toc503386196)

[Triangle 7](#_Toc503386197)

[Anibody.ui.\* 7](#_Toc503386198)

[Alert 7](#_Toc503386199)

[Confirm 7](#_Toc503386200)

[Prompt 7](#_Toc503386201)

[MultipleChoice 7](#_Toc503386202)

[ColorPicker 7](#_Toc503386203)

[Button 7](#_Toc503386204)

[InputField 7](#_Toc503386205)

[Slider 7](#_Toc503386206)

[Switch 7](#_Toc503386207)

[Toaster 7](#_Toc503386208)

[Anibody.util.\* 7](#_Toc503386209)

[PriorityQueue 7](#_Toc503386210)

[Timer 7](#_Toc503386211)

[IntervalHandler 7](#_Toc503386212)

[Counter 7](#_Toc503386213)

[Flow 7](#_Toc503386214)

[MultiFlow 7](#_Toc503386215)

[MediaManager 7](#_Toc503386216)

[Storage 7](#_Toc503386217)

[Task > Step 7](#_Toc503386218)

[Anibody.visual.\* 7](#_Toc503386219)

[ABOPresenter 7](#_Toc503386220)

[Animation 7](#_Toc503386221)

[ImageObject 7](#_Toc503386222)

[ABText 7](#_Toc503386223)

[CoordinateSystem 7](#_Toc503386224)

[Highlighting 7](#_Toc503386225)

[Spline 8](#_Toc503386226)

[Spotting 8](#_Toc503386227)

[Sprite 8](#_Toc503386228)

[Anibody.static.\* 8](#_Toc503386229)

[FAQ 8](#_Toc503386230)

# Preface

## Who wants to use Anibody?

Anyone who wants to bring his or her interactive animations or games to live and does not want to start from scratch. Anyone who wants to express them on the web and does not want to discriminate the viewer for what operation system or browser they use.

## Idea of the Engine

It provides the developer a possibility to implement game elements or interactive animations with less writing effort in a short period of time.

It is HTML5-conform and hence cross-browser friendly.

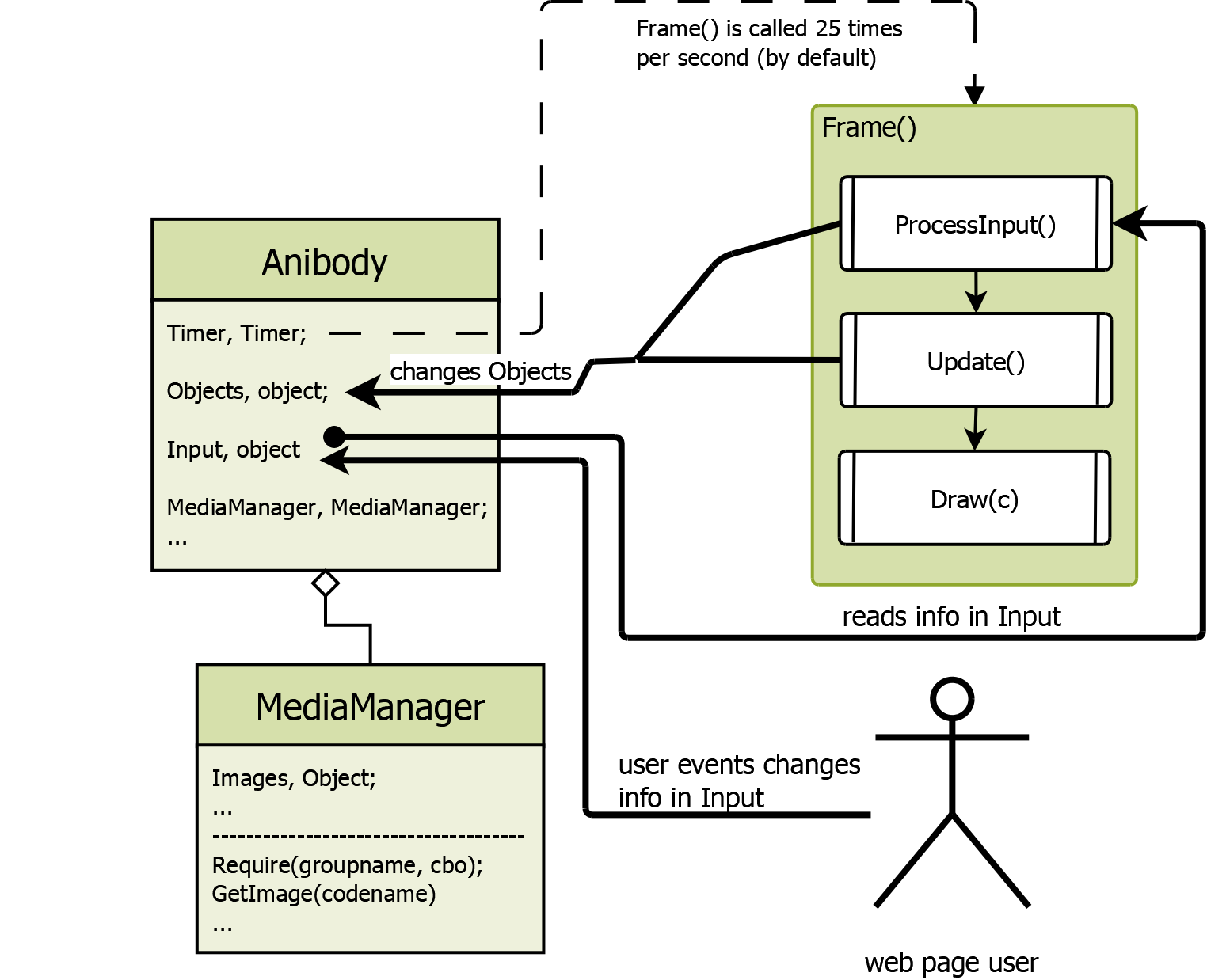
## Installation

|  |
| --- |
| <!-- You do not have to use jquery but if you do than jquery should be included before anibody -->  <script type="text/javascript" src="jqueryXYZ.js"></script>  <script type="text/javascript" src="anibodyXYZ.js"></script> |

## Basic Concept

All game related objects, which are registered for the Anibody engine, are going through the game cycle, when the Frame() method is called.

By default, a Timer calls the Frame() method 25 times per second – also known as FPS ( frames per second).



### The Game Cycle - here Frame()

**ProcessInput()**

In this phase of the game cycle, it is usual practice to read the information, which is saved in Anibody.Input, and manipulate the representative object or the environment according to your game logic.

For example: If the W key is pressed then move the representative game object 10 pixel upwards.

|  |
| --- |
| //this.Engine is a reference to the Anibody engine  if (this.Engine.Input.Keys.W.FramesPressed === 1) {  this.Move(0, -10);  } |

**Update()**

In this phase, if it is necessary to your game logic you can check the representative object or the general environment and handle the situation accordingly.

For example: If the mouse is over the graphical representation of this object, switch the cursor of the mouse to the “pointer”.

|  |
| --- |
| if(this.IsMouseOver)  this.Engine.Input.Mouse.Cursor.pointer(); |

Note: *At the end of ProcessInput(), the engine will resolve over which object the mouse is hovering, so now in the Update() you can safely check this.*

**Draw(c)**

In this phase every object with a graphical representation will be drawn on the canvases context (c variable).

# Documentation

List of Content

## Anibody

## Anibody.\*

### EngineObject

### ABO

### Widget

### DefaultCamera

### DefaultTerrain

## ECMAScriptExtension

## Anibody.debug.\*

### DumbObject

### Consolero

### DebugWindow

### Monitor

## Anibody.input.\*

- not the classes are important but the structure and the meaning of the data here

## Anibody.nav.\*

### BoxMenu

### Gallery

### SlideMenu > Tab

## Anibody.shapes.\*

### Shape

### Rectangle

### Circle

### Triangle

## Anibody.ui.\*

### Alert

### Confirm

### Prompt

### MultipleChoice

### ColorPicker

### Button

### InputField

### Slider

### Switch

### Toaster

## Anibody.util.\*

### PriorityQueue

### Timer

### IntervalHandler

### Counter

### Flow

### MultiFlow

### MediaManager

### Storage

### Task > Step

## Anibody.visual.\*

### ABOPresenter

### Animation

### ImageObject

### ABText

### CoordinateSystem

### Highlighting

### Spline

### Spotting

### Sprite

## Anibody.static.\*

# FAQ