

# Kaltura MediaSpace Styling Guide

---

Version: 5.0 - August 2016

---

**Kaltura Business Headquarters**

250 Park Avenue South, 10th Floor, New York, NY 10003

Tel.: +1 800 871 5224

---

Copyright © 2016 Kaltura Inc. All Rights Reserved. Designated trademarks and brands are the property of their respective owners.

Use of this document constitutes acceptance of the Kaltura Terms of Use and Privacy Policy.

---

# Contents

Preface .....	4
About this Guide .....	4
Audience .....	4
Document Conventions.....	4
Section 1 Kaltura MediaSpace Style Overview.....	5
Bootstrap CSS Resources .....	5
Section 2 Styling Guide.....	6
Header .....	6
Navigation .....	6
Page Titles .....	6
Gallery Action Bars .....	7
Buttons .....	7
Category/Channel Tabs .....	7
Thumbnails.....	7
Media Gallery Thumbnail .....	7
My Media Thumbnail.....	8
Channel Thumbnail .....	8
Tags .....	9
Appears In/Published In.....	9
Section 3 Best Practices for Using KMS Markup and the CSS .....	10
Modifying Core Elements Appearance and Behavior .....	11
Copy from Bootstrap.....	11
Choosing Your Selectors Wisely.....	11
When All Else Fails.....	12

# Preface

This preface contains the following topics:

- [About this Guide](#)
- [Audience](#)
- [Document Conventions](#)

## About this Guide

This guide describes how to create a custom CSS for KMS that modifies the core KMS Markup, and overrides the core KMS CSS.



**NOTE:** Please refer to the official and latest product release notes for last-minute updates.

Technical support may be obtained directly from: [Kaltura Customer Care](#).

### Contact Us:

Please send your documentation-related comments and feedback or report mistakes to [knowledge@kaltura.com](mailto:knowledge@kaltura.com). We are committed to improving our documentation and your feedback is important to us.

## Audience

This guide is intended for Kaltura Media Space Administrators and Developers.

## Document Conventions

Kaltura uses the following admonitions:

- Note
- Workflow



**NOTE:** Identifies important information that contains helpful suggestions.



**Workflow:** Provides workflow information.

1. Step 1
2. Step 2

# Kaltura MediaSpace Style Overview

Kaltura MediaSpace allows you to override the default style that is included in the application. The default style is defined in the bootstrap.css. Since the default theme in MediaSpace is built on top of the [Twitter Bootstrap framework](#), you can provide a standard [Bootstrap CSS](#) and override the default style that MediaSpace provides out-of-the-box.

This document defines the mark-up styles used through MediaSpace in each section so you can easily create your own custom CSS.

This document also provides best practices information for creating a custom CSS for KMS that modifies the core KMS Markup, overriding the core KMS CSS.

## Bootstrap CSS Resources

Before reading this guide, make sure you are familiar with the basic concepts and capabilities of Twitter Bootstrap:

- [Scaffolding](#) – Bootstrap is built on a responsive 12-column grid and [this](#) guide will explain the concepts and how it works.
- [Base CSS](#), [Components](#) and [JavaScript](#) - MediaSpace uses almost all of the standard Bootstrap components. This document references the appropriate component and class name for each section.

If you are familiar with CSS syntax, you can download the Bootstrap CSS and modify it in your preferred IDE, however, it is recommended that you use one of the following free tools to generate a standard Bootstrap CSS without the need to do any coding:

- [Lavish Bootstrap](#) – Generate your own Bootstrap color scheme from an image and customize to your taste.
- [BootSwatchr](#) - BootSwatchr is a visual tool for creating a custom Bootstrap theme from the ground up.
- [Bootswatcher](#) - Bootswatcher is an online tool that is used to create themes for Twitter Bootstrap with a live preview. You can create themes from scratch, from Bootswatch themes \and from images.
- [PaintStrap](#) - Generate colourful Twitter Bootstrap themes using the Adobe kuler / COLOURlovers color scheme.
- [StyleBootstrap.info](#) - Create a unique web design with Twitter Bootstrap.

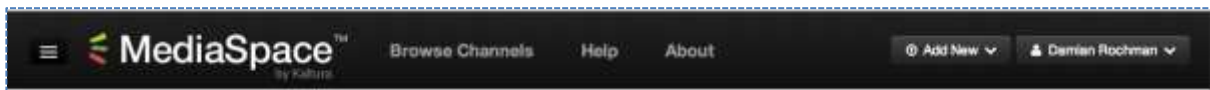
# Styling Guide

This section describes how you can style the MediaSpace user interface components.

## Header

The header style can be controlled both through the CSS file and a configuration in the KMS Admin Console. Depending on the option that is selected under *Header* → *headerStyle* (dark / light), the header will toggle the classes:

- **Dark** – `navbar-inverse` and `btn-inverse`

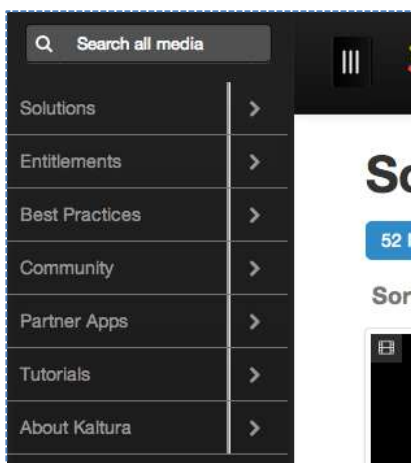


- **Light** – `navbar` and `btn`



## Navigation

Similar to the header section, the navigation section will toggle between `navbar-inverse` and `navbar` depending on the selected *headerStyle* selected.



## Page Titles

All page titles use the `h1` class.

# My Media

## Gallery Action Bars

Gallery action bars are used in different pages like My Media, Channel, etc. to group different actions related to a gallery of media items.

These bars use the `navbar`, `dropdown` and `dropdown-menu` classes.



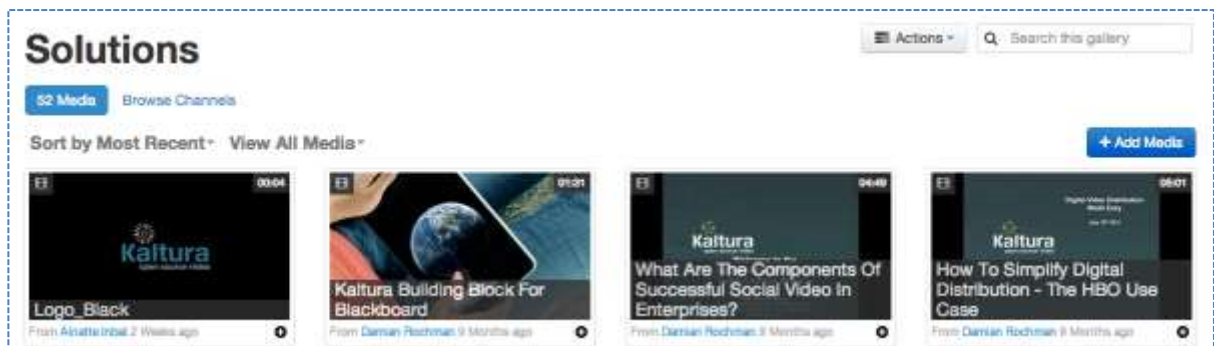
## Buttons

Buttons across different pages follow these conventions:

- Standard buttons use the `btn` class.
- Deletion buttons use the `btn-danger` class in addition to the `btn` class.
- Main / actionable buttons (like "Save", "Upload", etc) use the `btn-primary` class in addition to the `btn` class.
- Approve button (for moderated content) uses the `btn-success` class in addition to the `btn` class.
- Reject button (for moderated content) uses the `btn-danger` class in addition to the `btn` class.

## Category/Channel Tabs

In the category and channel pages there are multiple panes that are displayed (depending on the configuration) to display the media, channels or pending moderation items. These panes can be switched using the tabs in the page.



These tabs use the `tab` and `tab-pill` classes.

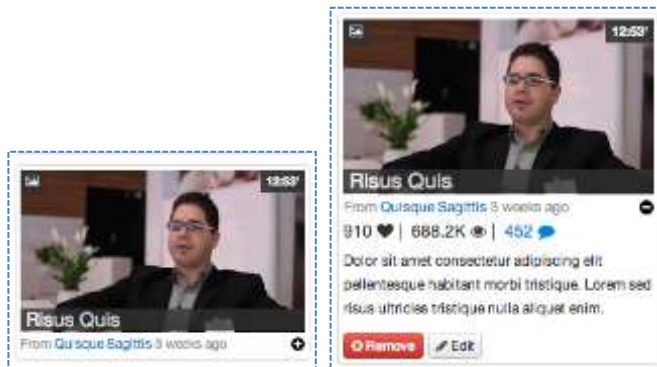
## Thumbnails

This section describes the different thumbnail configuration components.

### Media Gallery Thumbnail

## Styling Guide

The following applies to media thumbnails displayed in categories, channels, search pages and the related media panel.



Each thumbnail object is wrapped with the `thumbnail` class.

The labels on the thumbnail that show the icon with type of media, duration and name are all based on the `label-inverse` class.

The text that displays the user name and when the media was uploaded uses the `muted` class.

The description summary in the thumbnail uses the `small` class.

## My Media Thumbnail

The only difference in the My Media thumbnail and the Media Gallery thumbnail is that instead of the media name, The Media Gallery thumbnail displays a text bar with the status of the entry and its uses:

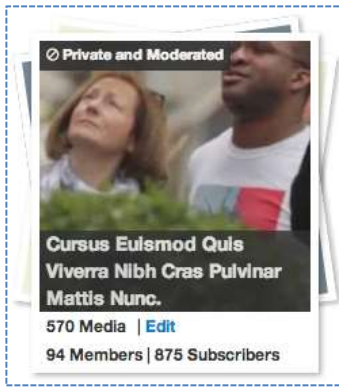
- `label-warning` for media items that are pending moderation at least in one category or channel.
- `label-info` for media items that are private (not published to any category or channel)



## Channel Thumbnail

Channel thumbnail labels use the `label-inverse` class as well.





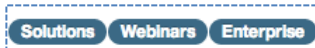
## Tags

Tags (keywords) of media are presented using `badge` classes and an `icon-tag` for the icon.



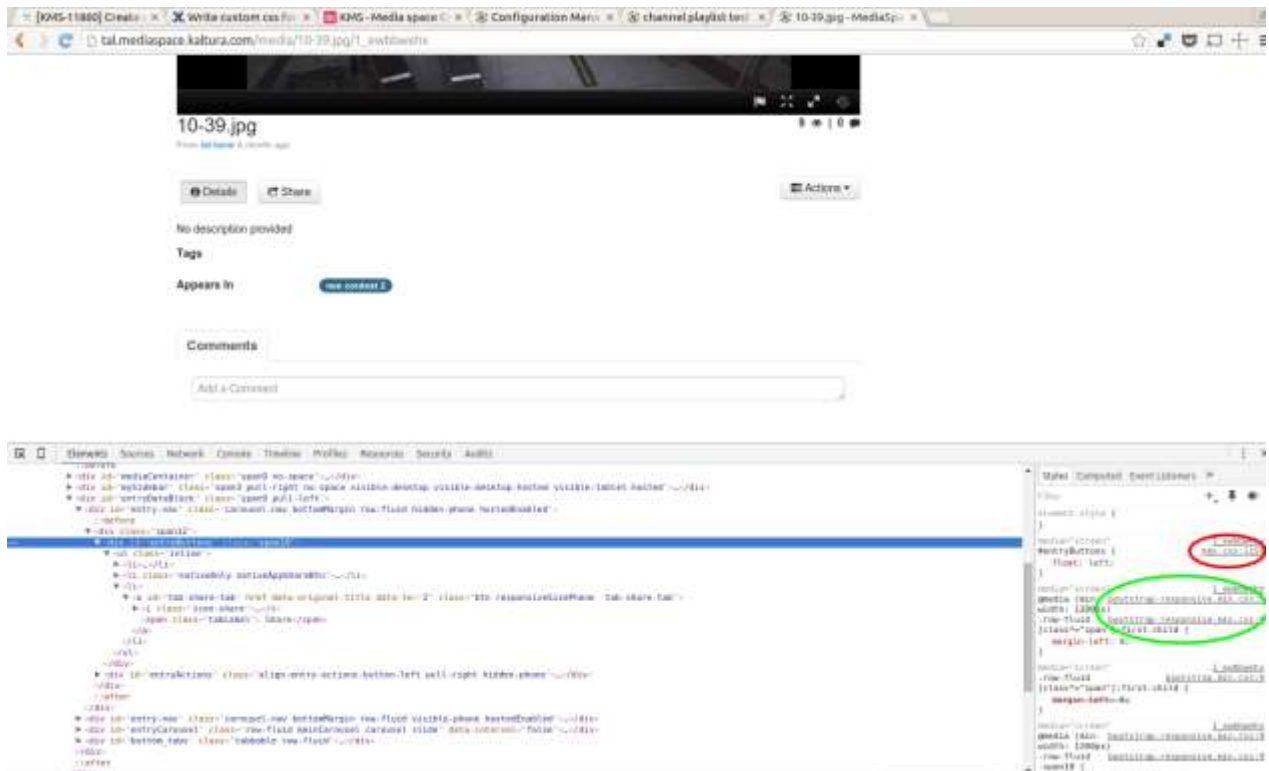
## Appears In/Published In

For each entry that is published to a category or channel, the published destination is presented using `badge` and `badge-info` classes (no icon).

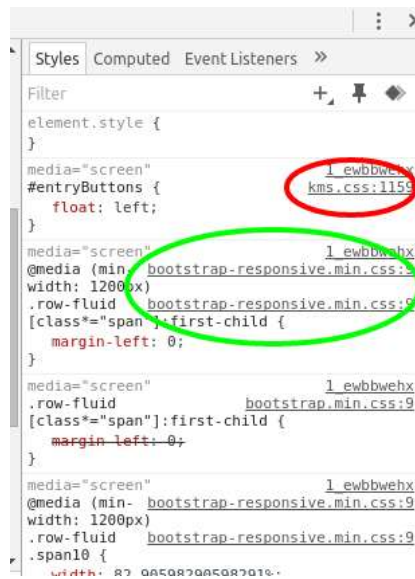


# Best Practices for Using KMS Markup and the CSS

The KMS 5 Markup and CSS are based on [bootstrap 2.3.2](#). All documentation pertaining to that version apply. KMS adds its own CSS to the mix. You can tell the origin of a given CSS directive via the Chrome Elements tab.



The class `.span10` came through bootstrap (see the green circle), but `#entryButtons` is a KMS originated id (see the red circle).



If you have `assetConsolidationEnabled` set to `true` in the Application module in your KMS admin, you will not be able to view the origin of a given CSS directive. We recommend turning this field off while developing code for KMS.

## Modifying Core Elements Appearance and Behavior

Since KMS markup is given to changes, it is challenging to modify the KMS core elements and their appearance and behaviour. KMS has many configuration options, making it hard to predict all the possible combinations of modules and their associated markup, and the interactions between them all.

### Copy from Bootstrap

If the behavior you want is covered by bootstrap, but lacks the element you need, you can copy the chosen CSS from the relevant bootstrap element.

For example, if you want your element to occupy a certain width, bootstraps `.spanN` classes have some CSSs that do just that.

The same goes for making buttons look like links - `btn btn-link` have the CSS you need.

It is best if you can use the bootstrap classes in your markup, but if circumstances prevent it, just copying the CSS itself will work.

## Choosing Your Selectors Wisely

Try and apply your CSS to elements with context on them.

1. `#id` is best. Element's ids do not change often.
2. Choose `.class` that is contextual, not bootstrap originated.  
For example - `.row-fluid` is a standard bootstrap class, and does not identify a specific element. `.channelSidebar` - is a contextual class, describing a specific KMS element.
3. When `#id`, or `.class` are not available - use the structure of the page with the least stages in between. This to avoid your selector being invalidated by changes to the existing markup. For example, if you want to apply css to a button under the header, use the minimal non-specific

css selector.

- `#header-container .btn` - will be valid as long as there is a button in the specific header-container.
- `.header .btn` - will be valid as long as there is a button in the header.
- `.header .row-fluid.pull-right .btn` - this selector will be invalidated if the page layout changes, and `.row-fluid` is not used any more.
- `.header > .row-fluid.pull-right > .btn` - this selector will be invalidated if elements are added between the `.header` and the `.btn` elements.

Take into account that elements can be added or removed.

- If you are using `:first` `:last` `:nth-child` - think carefully, will it always be the first/last/nth element? Or, can elements be added through configuration change?
- Are you certain that you are relating to the correct element? Or, can the element be enveloped in additional markup?

## When All Else Fails

In case you cannot not find a way of selecting the element you want to change, please contact Kaltura support with details. Kaltura will check the possibility of adding an ID either to the specific element, or an element in its route, which should provide a more direct selector to it.