Component Documentation

Textarea

A powerful multi-line text input component designed for collecting longer text content with comprehensive form integration, validation support, and accessibility features. Supports Angular Forms (reactive and template-driven), custom styling, and advanced features like processing gradient borders.

How to use

import { AavaTextareaComponent } from "@aava/play-core" ;

Basic Usage

Simple textarea implementations with labels and basic functionality.

```
<aava-textarea
  placeholder="Enter your text here..."
  [rows]="4"
  (change)="onTextareaChange($event)"
  [processing]="true"
  variant="primary"
></aava-textarea>

---

onTextareaChange(event: Event) {
    const target = event.target as HTMLTextAreaElement;
    console.log('Textarea value changed:', target.value);
  }
```

Sizes

Three size variants to accommodate different interface requirements and content needs.

```
<aava-textarea
  label="Extra Small"
  placeholder="Extra small textarea..."
  size="xs"
  [rows]="2"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  label="Small"
  placeholder="Small textarea..."
  size="sm"
  [rows]="2"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  label="Medium"
  placeholder="Medium textarea..."
  size="md"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  label="Large"
  placeholder="Large textarea..."
  size="lg"
  [rows]="4"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  label="Extra Large"
  placeholder="Extra large textarea..."
  size="xl"
  [rows]="4"
  (change)="onTextareaChange($event)"
></aava-textarea>

---

onTextareaChange(event: Event) {
    const target = event.target as HTMLTextAreaElement;
    console.log('Textarea value changed:', target.value);
  }
```

Available Sizes

- xs (Extra Small) : Very compact, ideal for tight spaces or dense forms
- sm (Small) : Compact for space-constrained interfaces
- md (Medium) : Standard size for most use cases (default)
- lg (Large) : Prominent for primary content input areas
- xl (Extra Large) : Very prominent, used for standout or high-visibility inputs

Variants

Color variants for different contexts, states, and semantic meanings.

```
<aava-textarea
  placeholder="Default variant..."
  variant="default"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  placeholder="Primary variant..."
  variant="primary"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  placeholder="Success variant..."
  variant="success"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  placeholder="Error variant..."
  variant="error"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  placeholder="Warning variant..."
  variant="warning"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<aava-textarea
  placeholder="Info variant..."
  variant="info"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

---

onTextareaChange(event: Event) {
    const target = event.target as HTMLTextAreaElement;
    console.log('Textarea value changed:', target.value);
  }
```

Style Variants

- default : Standard neutral appearance
- primary : Brand color for important inputs
- success : Green for positive confirmation states

- error : Red for validation errors and critical states
- warning : Orange for caution and attention states
- info : Blue for informational content and tips

Accessibility

Built-in accessibility features ensuring inclusive user experience.

```
<!-- Labeled Textarea -->
<aava-textarea
  label="Description"
  placeholder="Enter your description..."
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<!-- Textarea with Helper Text -->
<aava-textarea
  label="Comments"
  placeholder="Share your thoughts..."
  helper="Please provide detailed feedback to help us improve our service."
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<!-- Textarea with Error State -->
<aava-textarea
  label="Email"
  placeholder="Enter your email..."
  error="Please enter a valid email address."
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<!-- Textarea with Error State -->
<aava-textarea
  label="Required Field"
  placeholder="This field is required..."
  [required]="true"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

<!-- Textarea with Character Limit -->
<aava-textarea
  label="Bio"
  placeholder="Tell us about yourself (max 200 characters)..."
  [maxlength]="200"
  [rows]="3"
  (change)="onTextareaChange($event)"
></aava-textarea>

---

onTextareaChange(event: Event) {
    const target = event.target as HTMLTextAreaElement;
    console.log('Textarea value changed:', target.value);
  }
```

Accessibility Features

- Keyboard Navigation : Full keyboard support with proper focus management
- Screen Reader Support : Semantic HTML and ARIA attributes
- Error Announcements : Live regions for dynamic error messaging

- Focus Indicators : Clear visual focus states for navigation
- Label Association : Proper form control labeling and description
- High Contrast : Support for high contrast accessibility modes

API Reference

Inputs

Property	Type	Default	Description
label	string	''	Label text displayed above the textarea
placeholder	string	''	Placeholder text shown when textarea is empty
variant	'default' | 'primary' | 'success' | 'error' | 'warning' | 'info'	'default'	Color variant for styling
size	'xs' , 'sm' , 'md' , 'lg' , 'xl'	'md'	Size variant of the textarea
disabled	boolean	false	Whether the textarea is disabled
readonly	boolean	false	Whether the textarea is read-only
error	string	''	Error message to display
helper	string	''	Helper text to provide guidance
rows	number	3	Number of visible text lines
id	string	''	HTML id attribute for the textarea
name	string	''	HTML name attribute for form submission
maxlength	number	''	Maximum number of characters allowed
minlength	number	''	Minimum number of characters required
required	boolean	false	Whether the field is required

Property	Type	Default	Description
fullWidth	boolean	false	Whether to take full width of container
style	Record	''	Custom CSS styles object
resizable	boolean	true	Whether the textarea can be resized vertically
autoResize	boolean	false	Whether to automatically resize height to fit content
processing	boolean	false	Whether to show processing state
processingGradientBorder	boolean	false	Whether to show animated gradient border
processingGradientColors	string[]	See below	Array of colors for gradient animation

Outputs

Event	Type	Description
blur	EventEmitter	Emitted when the input loses focus
focus	EventEmitter	Emitted when the input gains focus
input	EventEmitter	Emitted on every input change
change	EventEmitter	Emitted when the input value changes
iconStartClick	EventEmitter	Emitted when the start icon is clicked
iconEndClick	EventEmitter	Emitted when the end icon is clicked

Properties

Property	Type	Description
value	string	Current textarea value
isFocused	boolean	Whether textarea currently has focus
hasError	boolean	Whether textarea has error state
hasHelper	boolean	Whether textarea has helper text

Methods

Method	Parameters	Return Type	Description
writeValue()	value: string	void	Set textarea value (ControlValueAccessor)
registerOnChange()	fn: (value: string) => void	void	Register change callback
registerOnTouched()	fn: () => void	void	Register touched callback
setDisabledState()	isDisabled: boolean	void	Set disabled state

CSS Custom Properties

Property	Description
--textbox-gap	Base gap between textarea elements
--textbox-gap-xs	Gap for extra small size variant
--textbox-gap-sm	Gap for small size variant
--textbox-gap-lg	Gap for large size variant
--textbox-gap-xl	Gap for extra large size variant
--textbox-label-font	Font family for labels
--textbox-label-color	Color for label text
--textbox-label-weight	Font weight for label text
--textbox-xs-label-font-size	Label font size for extra small variant
--textbox-xs-label-weight	Label font weight for extra small variant
--textbox-xs-label-line-height	Line height for extra small label

Property	Description
--textbox-xs-label-font-style	Font style for extra small label
--textbox-sm-label-font-size	Label font size for small variant
--textbox-sm-label-weight	Label font weight for small variant
--textbox-sm-label-line-height	Line height for small label
--textbox-sm-label-font-style	Font style for small label
--textbox-md-label-font-size	Label font size for medium variant
--textbox-md-label-weight	Label font weight for medium variant
--textbox-md-label-line-height	Line height for medium label
--textbox-md-label-font-style	Font style for medium label
--textbox-lg-label-font-size	Label font size for large variant
--textbox-lg-label-weight	Label font weight for large variant
--textbox-lg-label-line-height	Line height for large label
--textbox-lg-label-font-style	Font style for large label
--textbox-xl-label-font-size	Label font size for extra large variant
--textbox-xl-label-weight	Label font weight for extra large variant
--textbox-xl-label-line-height	Line height for extra large label
--textbox-xl-label-font-style	Font style for extra large label
--textbox-required-color	Color for required field indicator
--textbox-textarea-container-padding	Padding around textarea container
--textbox-background	Background color of the textarea
--glass-50-border	Glass border style for textarea
--textbox-textarea-hover-primary	Primary hover color for textarea
--textbox-textarea-hover-border-width	Border width on hover
--textbox-textarea-focus-primary	Primary color for focus state
--textbox-border-error-color	Border color when there is an error
--textbox-background-disabled	Background for disabled state
--textbox-border-disabled-color	Border color for disabled state
--textbox-background-readonly	Background for readonly state
--textbox-border-readonly-color	Border color for readonly state
--textbox-xs-border-radius	Border radius for extra small variant
--textbox-sm-border-radius	Border radius for small variant

Property	Description
--textbox-md-border-radius	Border radius for medium variant
--textbox-lg-border-radius	Border radius for large variant
--textbox-xl-border-radius	Border radius for extra large variant
--textbox-input-xs-padding	Padding inside extra small input
--textbox-input-sm-padding	Padding inside small input
--textbox-input-md-padding	Padding inside medium input
--textbox-input-lg-padding	Padding inside large input
--textbox-input-xl-padding	Padding inside extra large input
--textbox-input-font	Font family for input text
--textbox-input-color	Text color for input
--textbox-input-padding	General input padding
--textbox-textarea-min-height	Minimum height of textarea
--textbox-textarea-min-height-xs	Minimum height for extra small variant
--textbox-textarea-min-height-sm	Minimum height for small variant
--textbox-textarea-min-height-md	Minimum height for medium variant
--textbox-textarea-min-height-lg	Minimum height for large variant
--textbox-textarea-min-height-xl	Minimum height for extra large variant
--textbox-input-xs-font-size	Font size for extra small input
--textbox-input-xs-font-weight	Font weight for extra small input
--textbox-input-xs-font-style	Font style for extra small input
--textbox-input-xs-line-height	Line height for extra small input
--textbox-input-sm-font-size	Font size for small input
--textbox-input-sm-font-weight	Font weight for small input
--textbox-input-sm-font-style	Font style for small input
--textbox-input-sm-line-height	Line height for small input
--textbox-input-md-font-size	Font size for medium input
--textbox-input-md-font-weight	Font weight for medium input
--textbox-input-md-font-style	Font style for medium input
--textbox-input-md-line-height	Line height for medium input
--textbox-input-lg-font-size	Font size for large input
--textbox-input-lg-font-weight	Font weight for large input

Property	Description
--textbox-input-lg-font-style	Font style for large input
--textbox-input-lg-line-height	Line height for large input
--textbox-input-xl-font-size	Font size for extra large input
--textbox-input-xl-font-weight	Font weight for extra large input
--textbox-input-xl-font-style	Font style for extra large input
--textbox-input-xl-line-height	Line height for extra large input
--textbox-affix-padding	Padding for affix elements
--textbox-affix-color	Color of affix elements
--textbox-affix-font-size	Font size of affix elements
--textbox-affix-background	Background of affix elements
--textbox-affix-border-radius	Border radius of affix elements
--textbox-affix-disabled-color	Disabled color for affix elements
--textbox-affix-disabled-background	Disabled background for affix elements
--textbox-error-gap	Gap between textarea and error text
--textbox-error-color	Color for error messages
--textbox-error-font-size	Font size for error messages
--textbox-xs-error-font-size	Font size for error text in extra small variant
--textbox-sm-error-font-size	Font size for error text in small variant
--textbox-md-error-font-size	Font size for error text in medium variant
--textbox-lg-error-font-size	Font size for error text in large variant
--textbox-xl-error-font-size	Font size for error text in extra large variant
--textbox-helper-gap	Gap between textarea and helper text
--textbox-helper-color	Color for helper text
--textbox-helper-font-size	Font size for helper text
--textbox-xs-helper-font-size	Helper font size for extra small variant
--textbox-sm-helper-font-size	Helper font size for small variant
--textbox-md-helper-font-size	Helper font size for medium variant
--textbox-lg-helper-font-size	Helper font size for large variant
--textbox-xl-helper-font-size	Helper font size for extra large variant
--textbox-border-primary-color	Primary border color
--processing-gradient-colors	Colors for gradient animation

Best Practices

Design Guidelines

- Use appropriate row heights for expected content length
- Provide clear labels and helper text for guidance
- Choose semantic variants that match content purpose
- Enable resizing for longer content input areas
- Consider processing states for form submissions

Accessibility

- Always provide meaningful labels for screen readers
- Use helper text to explain input requirements
- Ensure error messages are descriptive and actionable
- Maintain sufficient color contrast for all variants
- Test keyboard navigation and screen reader compatibility

Performance

- Use OnPush change detection strategy for better performance
- Avoid frequent style changes that trigger repaints
- Consider debouncing for real-time validation
- Use trackBy functions for dynamic textarea lists