Consistent event handler parameters (eventmap-params)

Force consistent event handler parameters in event maps

Rule Details

Prevent the use of differently named parameters in event handlers to achieve more consistent code

The following patterns are considered warnings:

```
Template.foo.events({
  // 'foo' does not match 'event'
  'submit form': function (foo) {}
})

Template.foo.events({
  // 'bar' does not match 'templateInstance'
  'submit form': function (event, bar) {}
})

Template.foo.events({
  // neither 'foo' nor 'bar' are correct
  'submit form': function (foo, bar) {}
})
```

The following patterns are not warnings:

```
Template.foo.events({
  'submit form': function (event) {}
})

Template.foo.events({
  'submit form': function (event, templateInstance) {}
})

Template.foo.events({
  'submit form': function ({ target: form }, { data }) {}
})
```

Options

Parameter names

You can optionally set the names of the parameters.
You can set the name of the event parameter using `eventParamName` and the name of the template-instance parameter using `templateInstanceParamName`.
Here are examples of how to do this:

```
/*
 eslint meteor/eventmap-params: [2, {"eventParamName": "evt"}]
 */
Template.foo.events({
  'submit form': function (evt) {}
})

/*
 eslint meteor/eventmap-params: [2, {"templateInstanceParamName": "tmplInst"}]
 */
Template.foo.events({
  'submit form': function (event, tmplInst) {}
})

/*
 eslint meteor/eventmap-params: [2, {"eventParamName": "evt",
"templateInstanceParamName": "tmplInst"}]
 */
Template.foo.events({
  'submit form': function (evt, tmplInst) {}
})
```

Destructuring

You can optionally forbid destructuring the parameters.
You can set `preventDestructuring` to `"event"`, `"templateInstance"`, or `"both"`, to force no destructuring on the event parameter, template-instance parameter, or both respectively.

The following patterns are considered problems:

```
/*
 eslint meteor/eventmap-params: [2, {"preventDestructuring": "event"}]
 */
Template.foo.events({
  'submit form': function ({ target: form }, templateInstance) {}
})

/*
 eslint meteor/eventmap-params: [2, {"preventDestructuring": "templateInstance"}]
 */
Template.foo.events({
  'submit form': function (event, { data }) {}
})

/*
 eslint meteor/eventmap-params: [2, {"preventDestructuring": "both"}]
 */
Template.foo.events({
  'submit form': function (event, { data }) {}
})
```

The following patterns are not considered problems:

```
/*
 eslint meteor/eventmap-params: [2, {"preventDestructuring": "event"}]
 */
Template.foo.events({
  'submit form': function (event, { data }) {}
})

/*
 eslint meteor/eventmap-params: [2, {"preventDestructuring": "templateInstance"}]
 */
Template.foo.events({
  'submit form': function ({ target: form }, templateInstance) {}
})
```

Further Reading

- http://docs.meteor.com/#/full/eventmaps