Test Report with Mocha: Arlin Backend

All of these tests are passed. You can expect to see the actual and latest results on the CI/CD server (using Snap CI): https://snap-ci.com/gunadarma-academy/asde-michi-backend. Even more, in various pipelines of development, staging, and production.

TOC

# Feathers application tests starts and shows the index page.

```
request('http://localhost:3030', function(err, res, body) {
  assert.ok(body.indexOf('<html>') !== -1);
  done(err);
});
```

shows Arlin in title.

```
request('http://localhost:3030', function(err, res, body) {
  // expect(('<title>').to.have.html('Arlin'));
  done(err);
});
```

## error 404 shows a 404 HTML page.

```
request({
  url: 'http://localhost:3030/path/to/nowhere',
  headers: {
    'Accept': 'text/html'
  }
}, function(err, res, body) {
  assert.equal(res.statusCode, 404);
  assert.ok(body.indexOf('<html>') !== -1);
  done(err);
});
```

shows a 404 JSON error without stack trace.

```
request({
  url: 'http://localhost:3030/path/to/nowhere',
  json: true
}, function(err, res, body) {
  assert.equal(res.statusCode, 404);
  assert.equal(body.code, 404);
  assert.equal(body.message, 'Page not found');
  assert.equal(body.name, 'NotFound');
  done(err);
});
```

# message edit hook returns a function.

```
var hook = edit();
assert.equal(typeof hook, 'function');
```

filters data as expected.

```
const mockHook = {
  type: 'before',
  app: {},
  params: {},
  result: {},
  data: {
    text: 'arlin&'
  }
};
edit()(mockHook);
assert.deepEqual(mockHook.data, {
  text: 'arlin&amp;',
  updated_at: new Date().getTime()
});
```

# message init hook returns a function.

```
var hook = init();
assert.equal(typeof hook, 'function');
```

filters data as expected.

```
const mockHook = {
  type: 'before',
  app: {},
  params: {
    user: {
      _id: '123'
    }
  },
  result: {},
  data: {
    text: 'arlin&'
  }
};
init()(mockHook);
assert.deepEqual(mockHook.data, {
  text: 'arlin&amp;',
  author: '123',
  created_at: new Date().getTime()
});
```

# message restrict-to-sender hook returns a function.

```
var hook = restrictToSender();
assert.equal(typeof hook, 'function');
```

# message service registered the messages service.

```
assert.ok(app.service('messages'));
```

# question restrict-to-sender hook returns a function.

```
var hook = restrictToSender();
assert.equal(typeof hook, 'function');
```

# questions service registered the questions service.

```
assert.ok(app.service('questions'));
```

# quote service registered the quotes service.

```
assert.ok(app.service('quotes'));
```

# user gravatar hook returns a function.

```
var hook = gravatar();
assert.equal(typeof hook, 'function');
```

returns a gravatar url.

```
const mockHook = {
  type: 'before',
  app: {},
  params: {},
  result: {},
  data: {
    email: 'admin@arlin.link'
  }
};
gravatar()(mockHook);
assert.ok(mockHook.data.avatar);
```

# user service registered the users service.

```
assert.ok(app.service('users'));
```