

Codequalität

“ For every 1,000 lines of code, there are between 15 and 50 defects.

— Steve McConnell

# Was ist Codequalität?

Komplettheit

Verlässlichkeit

Wartbarkeit

Testbarkeit

Portabilität

Wiederverwendbarkeit

# Wartbarkeit

```
for j in range(len(numbers)):  
    for k in range(len(numbers) - 1):  
        if numbers[k] > numbers[k + 1]:  
            swap = numbers[k]  
            numbers[k] = numbers[k + 1]  
            numbers[k + 1] = swap
```

```
for j in range(len(numbers)):  
    for k in range(len(numbers) - 1):  
        if numbers[k] > numbers[k + 1]:  
            numbers[k] ^= numbers[k + 1]  
            numbers[k + 1] ^= numbers[k]  
            numbers[k] ^= numbers[k + 1]
```



**DRY**

- \* KISS
- \* SOLID
- \* YAGNI
- \* SOLID
- \* 90-90
- \* DTSTTCPW

# Style Guides



# ESLint

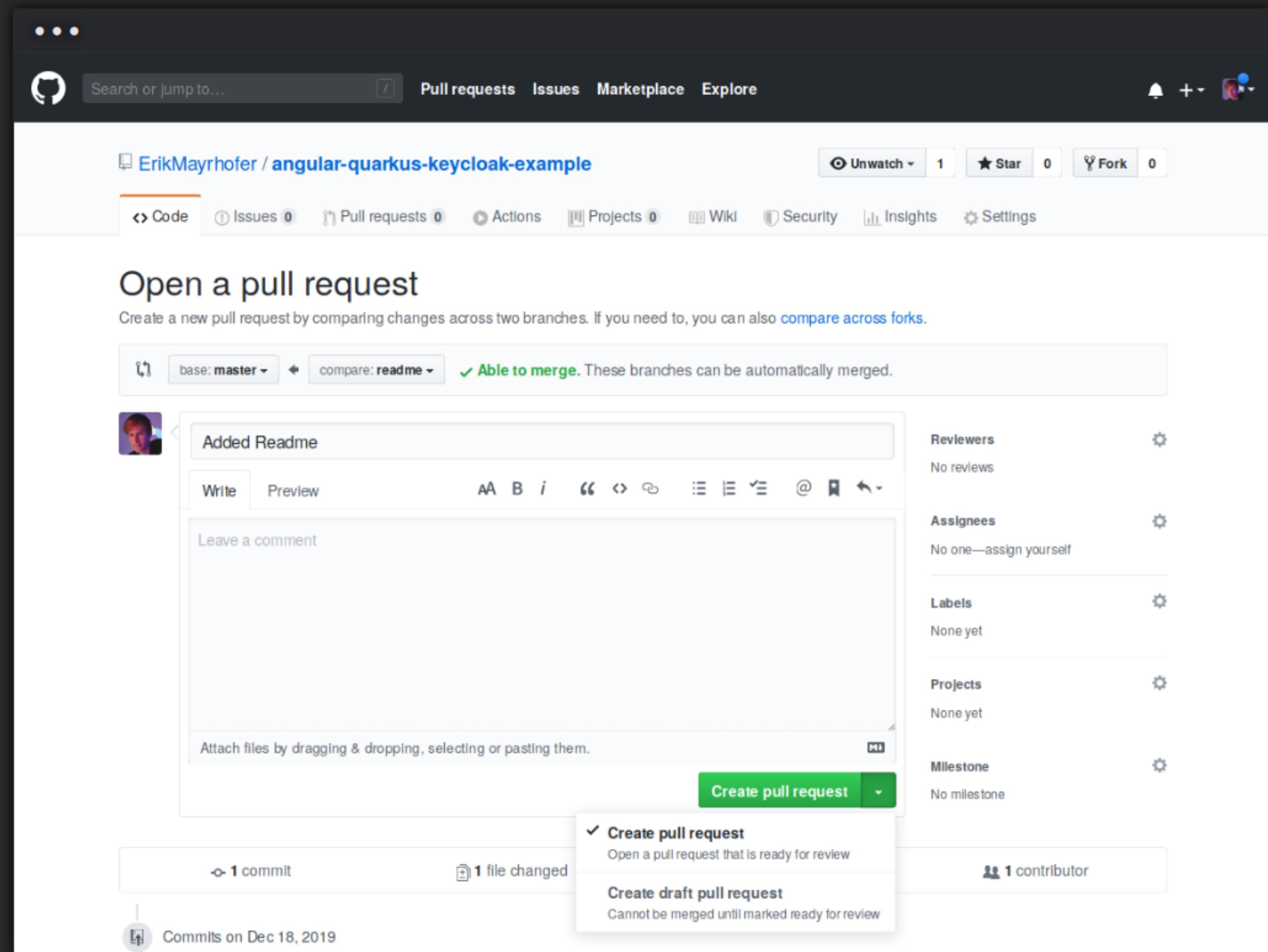
```
$> eslint myfile.js
✖ 53 problems (52 errors, 1 warning)
  49 errors and 0 warnings potentially fixable with the `--fix` option.

$> eslint myfile.js --fix _
```



# CodeReview

- \* Zeitbegrenzung
- \* Zeilenbegrenzung
- \* GitHub



Search or jump to... Pull requests Issues Marketplace Explore

ErikMayrhofer / angular-quarkus-keycloak-example Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 1 Actions Projects 0 Wiki Security Insights Settings

## Added Readme #1

**Open** ErikMayrhofer wants to merge 1 commit into `master` from `readme`

Conversation 0 Commits 1 Checks 0 Files changed 1 +9 -0

Changes from all commits File filter... Jump to... Review changes

Readme.adoc

```
+ = Keycloak Angular Quarkus
+ This is a Repository showing how to perform Authorization (OAuth2) using Keycloak, Angular and Quarkus.
+
+ == Structure
+
+ This Repository consists of three parts.
+
+ .`keycloak-quarkus-backend` is the Resource-Server implemented in Quarkus
+ .`web-app` contains the frontend, implemented in Angular
+ .`docker-compose.yml` contains the docker-image for the keycloak server.
```

ProTip! Use `n` and `p` to navigate between commits in a pull request.

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

# Documentation

- \* ReadTheDocs
- \* Slate
- \* Sphinx
- \* JavaDoc

# ReadTheDocs

The screenshot shows a desktop browser window displaying the Read the Docs website. The header includes the logo, a search bar, and navigation links for 'Docs' and 'Edit on GitHub'. The main content area is titled 'Read the Docs features' and contains several sections: 'GitHub, Bitbucket and GitLab Integration', 'Auto-updating', 'Internationalization', and 'Canonical URLs'. Each section includes a brief description and a link to more information. The sidebar on the left lists 'FIRST STEPS' and 'GETTING STARTED' sections, with 'Read the Docs features' expanded to show its sub-components.

Docs » Read the Docs features

Edit on GitHub

## Read the Docs features

This will serve as a list of all of the features that Read the Docs currently has. Some features are important enough to have their own page in the docs, others will simply be listed here.

### GitHub, Bitbucket and GitLab Integration

We now support linking by default in the sidebar. It links to the page on your host, which should help people quickly update typos and send pull requests to contribute to project documentation.

More information can be found in [Version Control System Integration](#).

### Auto-updating

The [Webhooks](#) page talks about the different ways you can ping RTD to let us know your project has been updated. We have official support for GitHub, Bitbucket and GitLab, and anywhere else we have a generic post-commit hook that allows you to POST to a URL to get your documentation built.

### Internationalization

Read the Docs itself is localized, and we support documentation translated into multiple languages. Read more on the [Localization of Documentation](#) and [Internationalization](#) pages.

### Canonical URLs

# Slate

The screenshot displays the Slate API documentation interface. On the left, a dark sidebar lists navigation items: Search, Introduction, **Authentication** (which is selected), Kittens, and Errors. Below the sidebar, there are links to "Sign Up for a Developer Key" and "Documentation Powered by Slate". The main content area has a light background. It starts with a section titled "Authentication" which explains API key usage and provides sample code for various languages (shell, ruby, python, javascript). The "Kittens" section follows, detailing the "Get All Kittens" endpoint, its HTTP request (GET http://example.com/api/kittens), and query parameters. A note at the bottom of the Kittens section states: "The above command returns JSON structured like this:".

**Authentication**

Kittn uses API keys to allow access to the API. You can register a new Kittn API key at our [developer portal](#).

Kittn expects for the API key to be included in all API requests to the server in a header that looks like the following:

```
Authorization: meowmeowmeow
```

**Info** You must replace `meowmeowmeow` with your personal API key.

**Kittens**

**Get All Kittens**

This endpoint retrieves all kittens.

**HTTP Request**

```
GET http://example.com/api/kittens
```

**Query Parameters**

To authorize, use this code:

```
require 'kittn'

api = Kittn::APIClient.authorize!("meowmeowmeow")
```

Make sure to replace `meowmeowmeow` with your API key.

```
require 'kittn'

api = Kittn::APIClient.authorize!("meowmeowmeow")
api.kittens.get
```

The above command returns JSON structured like this:

# Sphinx

The screenshot shows a Sphinx-generated documentation page for the Jinja templating engine. The page has a dark header bar with three dots on the left. Below the header, there's a sidebar on the left containing "Project Links" with links to "Donate to Pallets", "Jinja Website", "PyPI releases", "Source Code", and "Issue Tracker". There's also a "Quick search" input field with a "Go" button.

The main content area features the Jinja logo, which consists of a stylized Chinese character followed by the word "Jinja". Below the logo, a paragraph of text describes Jinja as a modern and designer-friendly templating language for Python, modeled after Django's templates. It is fast, widely used and secure with the optional sandboxed template execution environment.

Below the text, there's a code block showing a snippet of Jinja template code:

```
<title>{{ block title }}</title>
<ul>
  {% for user in users %}
    <li><a href="{{ user.url }}">{{ user.username }}</a></li>
  {% endfor %}
</ul>
```

Underneath the code block, there's a section titled "Features:" with a bulleted list:

- sandboxed execution
- powerful automatic HTML escaping system for XSS prevention
- template inheritance
- compiles down to the optimal python code just in time
- optional ahead-of-time template compilation
- easy to debug. Line numbers of exceptions directly point to the correct line in the template.
- configurable syntax

Below the features, there's a section titled "Contents:" with a hierarchical list:

- [Introduction](#)
  - [Prerequisites](#)
  - [Installation](#)
  - [Basic API Usage](#)
  - [Experimental Python 3 Support](#)
- [API](#)
  - [Basics](#)
  - [Unicode](#)
  - [High Level API](#)
  - [Autoescaping](#)
  - [Notes on Identifiers](#)
  - [Undefined Types](#)
  - [The Context](#)
  - [Loaders](#)
  - [Bytecode Cache](#)
  - [Async Support](#)
  - [Policies](#)

# JavaDoc

The screenshot shows the Java Platform, Standard Edition 7 API Specification documentation. The top navigation bar includes links for Overview, Package, Class, Use, Tree, Deprecated, Index, and Help. The main content area features a title section for "Java™ Platform, Standard Edition 7 API Specification" and a brief description stating it is the API specification for the Java™ Platform, Standard Edition. Below this is a "See: Description" link. A "Packages" section is highlighted, showing a table of packages and their descriptions:

Package	Description
<a href="#">java.applet</a>	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
<a href="#">java.awt</a>	Contains all of the classes for creating user interfaces and for painting graphics and images.
<a href="#">java.awt.color</a>	Provides classes for color spaces.
<a href="#">java.awt.datatransfer</a>	Provides interfaces and classes for transferring data between and within applications.
<a href="#">java.awt.dnd</a>	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
<a href="#">java.awt.event</a>	Provides interfaces and classes for dealing with different types of events. Final class AWT.

The left sidebar contains links for All Classes, Packages, and specific package lists for java.applet, java.awt, java.awt.color, and java.awt.datatransfer. The "All Classes" section is currently selected.

# JavaDoc

```
/**  
 * This method is used to add two integers. This is  
 * a the simplest form of a class method, just to  
 * show the usage of various javadoc Tags.  
 * @param numA This is the first paramter to addNum method  
 * @param numB This is the second parameter to addNum method  
 * @return int This returns sum of numA and numB.  
 */  
public int addNum(int numA, int numB) {  
    return numA + numB;  
}
```

# Testen

- \* Google Test
- \* JUnit
- \* Mocha
- \* Selenium

# CCI

- \* CodeClimate \$
- \* Codacy \$
- \* SonarQube

# CodeClimate

A screenshot of a GitHub repository page for 'RoboCop'. The top part shows a list of recent commits:

File	Commit Message	Date
LICENSE.txt	Update copyright years	6 years ago
README.md	Update redirects	4 years ago
Rakefile	Remove redundant 'rubocop' Rake task	4 years ago
rubocop.gemspec	Bump the parser dependency	4 years ago

The main content area features the RoboCop logo (a black helmet with a red visor) and the text "Role models are important. -- Officer Alex J. Murphy / RoboCop". Below the logo, there are badge links for gem version (0.77.0), FIX ME (Migrate to GitLab), build (unknown), coverage (80%), and maintainability (A).

# CodeClimate

The screenshot shows the CodeClimate dashboard for the repository `bbatsov/rubocop`. The top navigation bar includes links for Product, Pricing, Developers, Sign up, and Login. The main header displays the repository name `bbatsov/rubocop` with a star icon and a "Star" button. Below the header, there are tabs for Overview, Progress, Issues, Code, and Trends, with "Overview" being the active tab. A timestamp indicates the last build was 17 days ago.

**Breakdown**

**755 FILES**

Maintainability: (Full green)

Test coverage: (Full green)

**Codebase summary**

Maintainability	Test Coverage
1 mo	99%

**Repository stats**

Code Smells	Duplication	Other Issues
300	30	0

# Codacy

- \* Langsam
- \* Umfassend
- \* Security

# Codacy

My Repositories / angular-quarkus-keycloak-example Docs |

Issues breakdown

**32** total issues

Category	Total
Security	0
Error Prone	13
Code Style	19
Compatibility	0
Unused Code	0
Performance	0

[See all issues](#)

Coverage

Make sure your code is all tested. Set up your coverage here.

Analysis logs will show up here as soon as there are changes. [Learn more about logs](#).

Pull requests status

1 open pull requests

● Not up to standards 0  
● Up to standards 1

[See all pull requests](#)

# Codacy

The screenshot shows the Codacy interface for a repository named "angular-quarkus-keycloak-example". The left sidebar includes navigation links for Dashboard, Commits, Files, Issues (selected), Pull Requests, Security, Code patterns, and Settings. The main content area displays "Current Issues" for the "master" branch, filtered by "Kotlin". It lists three issues:

- keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/ExampleResource.kt**
  - UndocumentedPublicClass - [ExampleResource] at /src/keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/ExampleResource.kt:10:1
    - 11 @Path("/api")
  - UndocumentedPublicFunction - [resourceC] at /src/keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/ExampleResource.kt:24:5
    - 33 @GET
  - UndocumentedPublicClass - [ResourceThing] at /src/keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/ExampleResource.kt:30:1
    - 39 data class ResourceThing(val string: String)
- keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/UserSpecificResource.kt**
  - UndocumentedPublicClass - [UserSpecificResource] at /src/keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/UserSpecificResource.kt:15:1
    - 15 @Path("/api/user")
  - UndocumentedPublicFunction - [stuff] at /src/keycloak-quarkus-backend/src/main/kotlin/at/erikmayrhofer/UserSpecificResource.kt:30:5

# Codacy

The screenshot shows the Codacy web interface for a repository named "angular-quarkus-keycloak-example". The left sidebar includes navigation links for Dashboard, Commits, Files, Issues (selected), Pull Requests, Security, Code patterns, and Settings. The main content area displays "Current Issues" for the "master" branch, filtered by "Javascript". There are five listed issues, all of which are "Strings must use doublequote. (quotes)". The first issue is located in "web-app/e2e/protractor.conf.js":

```
5 const { SpecReporter } = require('jasmine-spec-reporter');
```

The second issue is also in "web-app/e2e/protractor.conf.js":

```
13 './src/**/*e2e-spec.ts'
```

The third issue is in "web-app/e2e/protractor.conf.js":

```
16 browserName: 'chrome'
```

The fourth issue is in "web-app/e2e/protractor.conf.js":

```
19 baseUrl: 'http://localhost:4200/>,
```

The fifth issue is in "web-app/e2e/protractor.conf.js":

```
20 framework: 'jasmine',
```

A small circular icon with a white envelope and a black border is visible in the bottom right corner of the main content area.

# Codacy

The screenshot shows the Codacy interface for a repository named "angular-quarkus-keycloak-example". The left sidebar includes links for Dashboard, Commits, Files, Issues, Pull Requests, Security (which is selected), and Settings. The main content area displays a list of "Warnings" under the "Security" category, with "Insecure Storage" highlighted in yellow. A tooltip indicates that enabling specific ESLint patterns is required for verification.

**Warnings**

- All >
- More >

- Auth >
- CSRF >
- Cryptography >
- DoS >
- File Access >
- Firefox OS >
- Input validation >
- Insecure Storage > **You need to enable the following patterns for this category to be verified**
  - Local Storage
    - Client-side data storage.  
ESLint
  - Session storage
    - Client-side data storage.  
ESLint
  - Local Storage
    - Client-side data storage.  
ESLint
  - Session Storage
    - Client-side data storage.  
ESLint
  - ESLint\_scanjs-rules\_property\_indexedDB
    - ESLint\_scanjs-rules\_property\_indexedDB  
ESLint
  - IndexedDB
    - Client-side data storage.  
ESLint
- Insecure modules/libraries >
- Other >
- Regex >
- Routes >
- SQL Injection >
- Unexpected behaviour >
- XSS >

# Codacy

The screenshot shows the Codacy interface for a repository named "angular-quarkus-keycloak-example". The left sidebar includes links for Dashboard, Commits, Files, Issues, Pull Requests, Security (which is selected), and Settings. The main content area displays a list of "Warnings" under the "Security" category, with "Insecure Storage" highlighted in yellow. A tooltip indicates that enabling specific ESLint patterns is required for verification.

**Warnings**

- All >
- More >

- Auth >
- CSRF >
- Cryptography >
- DoS >
- File Access >
- Firefox OS >
- Input validation >
- Insecure Storage > **You need to enable the following patterns for this category to be verified**
  - Local Storage
    - Client-side data storage.  
ESLint
  - Session storage
    - Client-side data storage.  
ESLint
  - Local Storage
    - Client-side data storage.  
ESLint
  - Session Storage
    - Client-side data storage.  
ESLint
  - ESLint\_scanjs-rules\_property\_indexedDB
    - ESLint\_scanjs-rules\_property\_indexedDB  
ESLint
  - IndexedDB
    - Client-side data storage.  
ESLint
- Insecure modules/libraries >
- Other >
- Regex >
- Routes >
- SQL Injection >
- Unexpected behaviour >
- XSS >

# SonarQube

```
version: "3"

services:
  sonarqube:
    image: sonarqube
    ports:
      - 9000:9000
```