




 [department-of-veterans-affairs](#) / [vets-website](#) Public

&lt;&gt; Code

 Issues 24 Pull requests 67 ZenHub Actions Projects 1 master ▾

...

[vets-website](#) / [.github](#) / [workflows](#) / [pull-request.yml](#)

ddzz Upgrade actions/setup-node to v3 (#20373) ✓

 History 5 contributors 264 lines (219 sloc) | 8.5 KB

...

```
1  name: Pull Request
2
3  on:
4    pull_request:
5      types: [ready_for_review, synchronize, reopened, opened]
6
7  env:
8    BOT_NAME: va-vfs-bot
9    PR_NUMBER: ${github.event.number}
10
11 jobs:
12   eslint-disable-check:
13     name: ESLint Disable Check
14     runs-on: ubuntu-latest
15     if: ${github.event.pull_request.draft == false}
16     steps:
17       - name: Checkout
18         uses: actions/checkout@v2
19
20       - name: Get Node version
21         id: get-node-version
22         run: echo ::set-output name=NODE_VERSION::$(cat .nvmrc)
23
24       - name: Setup Node
25         uses: actions/setup-node@v3
26         with:
27           node-version: ${steps.get-node-version.outputs.NODE_VERSION}
28
29       - name: Configure AWS credentials
```

[Sign in now](#) to use ZenHub

```
30     uses: aws-actions/configure-aws-credentials@v1
31     with:
32       aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID }
33       aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY }
34       aws-region: us-gov-west-1
35
36   - name: Get bot token from Parameter Store
37     uses: marvinpinto/action-inject-ssm-secrets@v1.2.1
38     with:
39       ssm_parameter: /devops/VA_VFS_BOT_GITHUB_TOKEN
40       env_variable_name: GITHUB_TOKEN
41
42   - name: Cache dependencies
43     id: cache-dependencies
44     uses: actions/cache@v2
45     with:
46       path: |
47         ~/.cache/yarn
48         node_modules
49       key: ${ steps.get-node-version.outputs.NODE_VERSION }-${ hashFiles('yarn.lock') }
50       restore-keys: ${ steps.get-node-version.outputs.NODE_VERSION }-
51
52   - name: Install dependencies
53     uses: nick-invision/retry@v2
54     with:
55       command: yarn install --frozen-lockfile --prefer-offline
56       max_attempts: 3
57       timeout_minutes: 7
58     env:
59       YARN_CACHE_FOLDER: ~/.cache/yarn
60
61   - name: Run PR check script
62     run: yarn pr-check
63     env:
64       CODE_PATTERN: (/* eslint-disable)|(// eslint-disable)
65       LINE_COMMENT: ESLint disabled here
66       OVERALL_REVIEW_COMMENT: >
67         # ESLint is disabled
68
69         `vets-website` uses ESLint to help enforce code quality. In most
70         situations we would like ESLint to remain enabled.
71
72         ## What you can do
73
74         See if the code can be refactored to avoid disabling ESLint, or wait for
75         a VSP review.
76
77   icon-check:
78     name: Icon Check
```

[Sign in now](#) to use ZenHub

```
79 runs-on: ubuntu-latest
80 if: ${ github.event.pull_request.draft == false }}
81 steps:
82   - name: Checkout
83     uses: actions/checkout@v2
84
85   - name: Get Node version
86     id: get-node-version
87     run: echo ::set-output name=NODE_VERSION::$(cat .nvmrc)
88
89   - name: Setup Node
90     uses: actions/setup-node@v3
91     with:
92       node-version: ${ steps.get-node-version.outputs.NODE_VERSION }}
93
94   - name: Configure AWS credentials
95     uses: aws-actions/configure-aws-credentials@v1
96     with:
97       aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID }}
98       aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY }}
99       aws-region: us-gov-west-1
100
101   - name: Get bot token from Parameter Store
102     uses: marvinpinto/action-inject-ssm-secrets@v1.2.1
103     with:
104       ssm_parameter: /devops/VA_VFS_BOT_GITHUB_TOKEN
105       env_variable_name: GITHUB_TOKEN
106
107   - name: Cache dependencies
108     id: cache-dependencies
109     uses: actions/cache@v2
110     with:
111       path: |
112         ~/.cache/yarn
113         node_modules
114       key: ${ steps.get-node-version.outputs.NODE_VERSION }}-${ hashFiles('yarn.lock') }
115       restore-keys: ${ steps.get-node-version.outputs.NODE_VERSION }}-
116
117   - name: Install dependencies
118     uses: nick-invision/retry@v2
119     with:
120       command: yarn install --frozen-lockfile --prefer-offline
121       max_attempts: 3
122       timeout_minutes: 7
123     env:
124       YARN_CACHE_FOLDER: ~/.cache/yarn
125
126   - name: Run PR check script
127     run: yarn pr-check
```

[Sign in now](#) to use ZenHub

```
128   env:
129     CODE_PATTERN: (<i )|(<i$)
130     LINE_COMMENT: icon found
131     OVERALL_REVIEW_COMMENT: >
132       # Icon found
133
134       Icons can be decorative, but sometimes they are used to convey meaning.
135       If there are any semantics associated with an icon, those semantics
136       should also be conveyed to a screen reader.
137
138       ## What you can do
139
140       Review the markup and see if the icon provides information that isn't
141       represented textually, or wait for a VSP review.
142
143 sentry-check:
144   name: Sentry Check
145   runs-on: ubuntu-latest
146   if: ${ github.event.pull_request.draft == false }
147   steps:
148     - name: Checkout
149       uses: actions/checkout@v2
150
151     - name: Get Node version
152       id: get-node-version
153       run: echo ::set-output name=NODE_VERSION::$(cat .nvmrc)
154
155     - name: Setup Node
156       uses: actions/setup-node@v3
157       with:
158         node-version: ${ steps.get-node-version.outputs.NODE_VERSION }
159
160     - name: Configure AWS credentials
161       uses: aws-actions/configure-aws-credentials@v1
162       with:
163         aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID }
164         aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY }
165         aws-region: us-gov-west-1
166
167     - name: Get bot token from Parameter Store
168       uses: marvinpinto/action-inject-ssm-secrets@v1.2.1
169       with:
170         ssm_parameter: /devops/VA_VFS_BOT_GITHUB_TOKEN
171         env_variable_name: GITHUB_TOKEN
172
173     - name: Cache dependencies
174       id: cache-dependencies
175       uses: actions/cache@v2
176       with:
```

[Sign in now](#) to use ZenHub

```
177     path: |
178         ~/.cache/yarn
179         node_modules
180     key: ${ steps.get-node-version.outputs.NODE_VERSION }}-${ hashFiles('yarn.lock')
181     restore-keys: ${ steps.get-node-version.outputs.NODE_VERSION }}-
182
183 - name: Install dependencies
184   uses: nick-invision/retry@v2
185   with:
186     command: yarn install --frozen-lockfile --prefer-offline
187     max_attempts: 3
188     timeout_minutes: 7
189   env:
190     YARN_CACHE_FOLDER: ~/.cache/yarn
191
192 - name: Run PR check script
193   run: yarn pr-check
194   env:
195     CODE_PATTERN: Sentry\
196     LINE_COMMENT: Sentry found
197     OVERALL_REVIEW_COMMENT: >
198       # Sentry call found
199
200       Sentry captures a lot of data, and we want to make sure that we only
201       keep information that will be useful for troubleshooting issues. This
202       means that PII should not be recorded.
203
204       ## What you can do
205
206       Review your call to Sentry and see if you can reasonably reduce any
207       information that is included, or wait for a VSP review.
208
209 linting:
210   name: Linting (Files Changed)
211   runs-on: ubuntu-latest
212   steps:
213     - name: Checkout
214       uses: actions/checkout@v2
215
216     - name: Get Node version
217       id: get-node-version
218       run: echo ::set-output name=NODE_VERSION::$(cat .nvmrc)
219
220     - name: Setup Node
221       uses: actions/setup-node@v3
222       with:
223         node-version: ${ steps.get-node-version.outputs.NODE_VERSION }}
224
225     - name: Cache dependencies
```

[Sign in now](#) to use ZenHub

```
226     id: cache-dependencies
227     uses: actions/cache@v2
228     with:
229       path: |
230         ~/.cache/yarn
231         node_modules
232       key: ${ steps.get-node-version.outputs.NODE_VERSION }}-${ hashFiles('yarn.lock
233       restore-keys: ${ steps.get-node-version.outputs.NODE_VERSION }}-
234
235 - name: Install dependencies
236   uses: nick-invision/retry@v2
237   with:
238     command: yarn install --frozen-lockfile --prefer-offline
239     max_attempts: 3
240     timeout_minutes: 7
241   env:
242     YARN_CACHE_FOLDER: ~/.cache/yarn
243
244 - name: Get files changed
245   id: get-files-changed
246   uses: trilom/file-changes-action@v1.2.4
247   continue-on-error: true
248   with:
249     output: 'json'
250
251 - name: Filter files
252   id: get-lint-files
253   run: node ./script/github-actions/filter-files-changed.js ${ steps.get-files-char
254
255 - name: Annotate ESLint results
256   run: yarn run eslint -c ./eslintrc.changed.js --ext .js --ext .jsx --no-error-on-
257
258 - name: Run Stylelint
259   if: ${ steps.get-lint-files.outputs.SCSSFILES != '' }}
260   run: yarn run stylelint verbose --output-file stylelint-report.json --formatter js
261
262 - name: Annotate Stylelint results
263   if: ${ always() }}
264   run: node ./script/github-actions/stylelint-annotation-format.js
```