```
Started by user admin
Running in Durability level: MAX_SURVIVABILITY
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/jobs/checkbox.io/workspace
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Source)
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/chrisparnin/checkbox.io # timeout=10
Fetching upstream changes from https://github.com/chrisparnin/checkbox.io
 > git --version # timeout=10
 > git --version # 'git version 2.25.1'
 > git fetch --tags --force --progress -- https://github.com/chrisparnin/checkbox.io +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision bfbf426fbc55de9139decf152169a7ae72839e76 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f bfbf426fbc55de9139decf152169a7ae72839e76 # timeout=10
 > git branch -a -v --no-abbrev # timeout=10
 > git branch -D master # timeout=10
 > git checkout -b master bfbf426fbc55de9139decf152169a7ae72839e76 # timeout=10
Commit message: "Update longmethod.js"
 > git rev-list --no-walk bfbf426fbc55de9139decf152169a7ae72839e76 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
Installing node modules
[Pipeline] sh
+ cd server-side/site
+ npm install
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.2 (node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.2: wanted {"os":"darwin","arch":"any"} (current:
{"os":"linux","arch":"x64"})

audited 338 packages in 6.694s

16 packages are looking for funding
  run `npm fund` for details

found 38 vulnerabilities (22 low, 3 moderate, 13 high)
  run `npm audit fix` to fix them, or `npm audit` for details
[Pipeline] echo
```

```
Starting mongodb
[Pipeline] sh
+ sudo service mongod start
[Pipeline] echo
Starting node server
[Pipeline] sh
+ cd server-side/site
+ pm2 start server.js
[PM2] Spawning PM2 daemon with pm2_home=/var/lib/jenkins/.pm2
[PM2] PM2 Successfully daemonized
[PM2] Starting /var/lib/jenkins/jobs/checkbox.io/workspace/server-side/site/server.js in fork_mode (1 instance)
[PM2] Done.
┌─────┬───────────┬─────────────┬─────────┬─────────┬──────────┬────────┬──────┬───────────┬──────────┬──────────┬──────────┬──────────
─┐
│ id  │ name      │ namespace   │ version │ mode    │ pid      │ uptime │ ↺    │ status    │ cpu      │ mem      │ user     │ watching
│
├─────┼───────────┼─────────────┼─────────┼─────────┼──────────┼────────┼──────┼───────────┼──────────┼──────────┼──────────┼──────────
─┤
│ 0   │ server    │ default     │ 0.0.1   │ fork    │ 11709    │ 0s     │ 0    │ online    │ 0%       │ 34.2mb   │ jenkins  │ disabled
│
└─────┴───────────┴─────────────┴─────────┴─────────┴──────────┴────────┴──────┴───────────┴──────────┴──────────┴──────────┴──────────
─┘
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] sh
+ cd server-side/site
+ npm test

> design-node@0.0.1 test /var/lib/jenkins/jobs/checkbox.io/workspace/server-side/site
> mocha



(node:11752) Warning: Accessing non-existent property 'count' of module exports inside circular dependency
(Use `node --trace-warnings ...` to show where the warning was created)
(node:11752) Warning: Accessing non-existent property 'findOne' of module exports inside circular dependency
(node:11752) Warning: Accessing non-existent property 'remove' of module exports inside circular dependency
(node:11752) Warning: Accessing non-existent property 'updateOne' of module exports inside circular dependency
  GET /api/study/listing
connected!
[]
(node:11752) [DEP0066] DeprecationWarning: OutgoingMessage.prototype._headers is deprecated
    ✓ responds with json (502ms)

  Render markdown
{NumberQuestions:true}
<p><div data-page="0"></p>
```

```
<p>Start with header for global options:</p>
<pre><code>{NumberQuestions:true}
-----------</code></pre><p><div data-question="1" data-kind="multichoice"></p>
<h3 id="1-multiple-choice-question-check-all-that-apply">1. Multiple Choice Question (Check all that apply)</h3>
<p>A <em>description</em> for question.<br>Questions are created with headers (level 3) <code>### Multiple Choice Question (Check all
that apply)</code>.</p>
<p>
<ul>
  <li>
    <div class="controls">
      <label class="checkbox">Choice A
        <input type="checkbox" name="#{choice.name}" value="#{choice.value}"/>
      </label>
    </div>
  </li>
  <li>
    <div class="controls">
      <label class="checkbox">Choice B
        <input type="checkbox" name="#{choice.name}" value="#{choice.value}"/>
      </label>
    </div>
  </li>
  <li>
    <div class="controls">
      <label class="checkbox">Choice C
        <input type="checkbox" name="#{choice.name}" value="#{choice.value}"/>
      </label>
    </div>
  </li>
</ul>
</div>
<div data-question="2" data-kind="singlechoice"></p>
<h3 id="2-single-choice-question">2. Single Choice Question</h3>
<p>Markdown is great for including questions about code snippets:</p>
<pre><code>$(document).ready( function()
{
    ko.applyBindings(new TaskViewModel());
    var converter = new Markdown.Converter();
    document.write(converter.makeHtml(&quot;**I am bold**&quot;));
});</code></pre><p>
<ol>
  <li>
    <div class="controls">
      <label class="radio">Choice
        <input type="radio" name="#{choice.name}" value="#{choice.value}"/>
      </label>
    </div>
  </li>
  <li>
```

```
    <div class="controls">
      <label class="radio">Choice
        <input type="radio" name="#{choice.name}" value="#{choice.value}"/>
      </label>
    </div>
  </li>
  <li>
    <div class="controls">
      <label class="radio">Choice
        <input type="radio" name="#{choice.name}" value="#{choice.value}"/>
      </label>
    </div>
  </li>
</ol>
</div>
<div data-question="3" data-kind="singlechoicetable"></p>
<h3 id="3-rankingrating-table">3. Ranking/Rating Table</h3>
<p>The first column has the description.  <a href="https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet#wiki-tables">Use github flavored markdown for table formatting</a>.</p>
<p>
<table class="questionTable table-bordered table-condensed">
  <thead>
    <tr>
      <th></th>
      <th>Do not Want</th>
      <th>Sometimes</th>
      <th>Always</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Search terms used in IDE</td>
      <td>
        <div class="surveycell">
          <label class="radio">
            <input type="radio" value="#{c}" name="#{table}_#{r}"/>
          </label>
        </div>
      </td>
      <td>
        <div class="surveycell">
          <label class="radio">
            <input type="radio" value="#{c}" name="#{table}_#{r}"/>
          </label>
        </div>
      </td>
      <td>
        <div class="surveycell">
          <label class="radio">
```

```
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
      </tr>
      <tr>
        <td>Code that did not work out and was deleted.</td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
      </tr>
      <tr>
        <td>Time spent on particular edits</td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
      </tr>
      <tr>
        <td>Code and files viewed</td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
        <td>
          <div class="surveycell">
            <label class="radio">
              <input type="radio" value="#{c}" name="#{table}_#{r}"/>
            </label>
          </div>
        </td>
      </tr>
    </tbody>
  </table>
  </div></p>

      ✓ successfully renders survey


    2 passing (542ms)

  [Pipeline] }
  [Pipeline] // stage
  [Pipeline] stage
  [Pipeline] { (Tear-down)
  [Pipeline] sh
  + cd server-side/site
  + pm2 stop server.js
  [PM2] Applying action stopProcessId on app [server.js](ids: [ 0 ])
  [PM2] [server](0) ✓
  ┌─────┬────────────┬─────────────┬─────────┬─────────┬──────────┬────────┬──────┬───────────┬──────────┬──────────┬──────────┬──────────
  ┐
  │ id  │ name       │ namespace   │ version │ mode    │ pid      │ uptime │ ↺    │ status    │ cpu      │ mem      │ user     │ watching
```

```
│
├─────┼───────────┼─────────────┼─────────┼─────────┼──────────┼────────┼──────┼───────────┼──────────┼──────────┼──────────┼─────────
─┤
│ 0   │ server    │ default     │ 0.0.1   │ fork    │ 0        │ 0      │ 0    │ stopped   │ 0%       │ 0b       │ jenkins  │ disabled
│
└─────┴───────────┴─────────────┴─────────┴─────────┴──────────┴────────┴──────┴───────────┴──────────┴──────────┴──────────┴─────────
─┘
[Pipeline] sh
+ sudo service mongod stop
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```