

v0.12.10 ▾

...

gogs / public / js / gogs.js / <> Jump to ▾



alpenrossi repo: disable public access option when using external wiki and iss... ... ✖

🕒 History

👤 17 contributors



+5

1798 lines (1654 sloc) | 46.7 KB

...

```
1  "use strict";
2
3  var csrf;
4  var suburl;
5
6  function initCommentPreviewTab($form) {
7      var $tabMenu = $form.find(".tabular.menu");
8      $tabMenu.find(".item").tab();
9      $tabMenu
10         .find('.item[data-tab="' + $tabMenu.data("preview") + '"']')
11         .click(function() {
12             var $this = $(this);
13             $.post(
14                 $this.data("url"),
15                 {
16                     _csrf: csrf,
17                     mode: "gfm",
18                     context: $this.data("context"),
19                     text: $form
20                         .find(
21                             '.tab.segment[data-tab="' + $tabMenu.data("write") + '"'] textarea'
22                         )
23                     .val()
24                 },
25             function(data) {
26                 var $previewPanel = $form.find(
27                     '.tab.segment[data-tab="' + $tabMenu.data("preview") + '"']'
28                 );
```

```

29     $previewPanel.html(data);
30     emojiify.run($previewPanel[0]);
31     $("pre code", $previewPanel[0]).each(function(i, block) {
32         hljs.highlightBlock(block);
33     });
34     }
35     );
36     });
37
38     buttonsClickOnEnter();
39 }
40
41 var previewFileModes;
42
43 function initEditPreviewTab($form) {
44     var $tabMenu = $form.find(".tabular.menu");
45     $tabMenu.find(".item").tab();
46     var $previewTab = $tabMenu.find(
47         '.item[data-tab="' + $tabMenu.data("preview") + '"]'
48     );
49     if ($previewTab.length) {
50         previewFileModes = $previewTab.data("preview-file-modes").split(",");
51         $previewTab.click(function() {
52             var $this = $(this);
53             $.post(
54                 $this.data("url"),
55                 {
56                     _csrf: csrf,
57                     context: $this.data("context"),
58                     text: $form
59                         .find(
60                             '.tab.segment[data-tab="' + $tabMenu.data("write") + '"] textarea'
61                         )
62                         .val()
63                 },
64                 function(data) {
65                     var $previewPanel = $form.find(
66                         '.tab.segment[data-tab="' + $tabMenu.data("preview") + '"]'
67                     );
68                     $previewPanel.html(data);
69                     emojiify.run($previewPanel[0]);
70                     $("pre code", $previewPanel[0]).each(function(i, block) {
71                         hljs.highlightBlock(block);
72                     });
73                 }
74             );
75         });
76     }
77 }

```

```

78
79 function initEditDiffTab($form) {
80     var $tabMenu = $form.find(".tabular.menu");
81     $tabMenu.find(".item").tab();
82     $tabMenu
83         .find('.item[data-tab="' + $tabMenu.data("diff") + '"']')
84         .click(function() {
85             var $this = $(this);
86             $.post(
87                 $this.data("url"),
88                 {
89                     _csrf: csrf,
90                     content: $form
91                         .find(
92                             '.tab.segment[data-tab="' + $tabMenu.data("write") + '"'] textarea'
93                         )
94                         .val()
95                 },
96                 function(data) {
97                     var $diffPreviewPanel = $form.find(
98                         '.tab.segment[data-tab="' + $tabMenu.data("diff") + '"']'
99                     );
100                     $diffPreviewPanel.html(data);
101                     emojiify.run($diffPreviewPanel[0]);
102                 }
103             );
104         });
105 }
106
107 function initEditForm() {
108     if ($("#edit.form").length == 0) {
109         return;
110     }
111
112     initEditPreviewTab($("#edit.form"));
113     initEditDiffTab($("#edit.form"));
114 }
115
116 function initCommentForm() {
117     if ($("#comment.form").length == 0) {
118         return;
119     }
120
121     initCommentPreviewTab($("#comment.form"));
122
123     // Labels
124     var $list = $(".ui.labels.list");
125     var $noSelect = $list.find(".no-select");
126     var $labelMenu = $(".select-label .menu");

```

```

127     var hasLabelUpdateAction = $labelMenu.data("action") == "update";
128
129     function updateIssueMeta(url, action, id) {
130         $.post(url, {
131             _csrf: csrf,
132             action: action,
133             id: id
134         });
135     }
136
137     // Add &nbsp; to each unselected label to keep UI looks good.
138     // This should be added directly to HTML but somehow just get empty <span> on this page.
139     $labelMenu
140         .find(".item:not(.no-select) .octicon:not(.octicon-check)")
141         .each(function() {
142             $(this).html("&nbsp;");
143         });
144     $labelMenu.find(".item:not(.no-select)").click(function() {
145         if ($(this).hasClass("checked")) {
146             $(this).removeClass("checked");
147             $(this)
148                 .find(".octicon")
149                 .removeClass("octicon-check")
150                 .html("&nbsp;");
151             if (hasLabelUpdateAction) {
152                 updateIssueMeta(
153                     $labelMenu.data("update-url"),
154                     "detach",
155                     $(this).data("id")
156                 );
157             }
158         } else {
159             $(this).addClass("checked");
160             $(this)
161                 .find(".octicon")
162                 .addClass("octicon-check")
163                 .html("");
164             if (hasLabelUpdateAction) {
165                 updateIssueMeta(
166                     $labelMenu.data("update-url"),
167                     "attach",
168                     $(this).data("id")
169                 );
170             }
171         }
172     }
173
174     var labelIds = "";
175     $(this)
176         .parent()

```

```

176     .find(".item")
177     .each(function() {
178         if ($(this).hasClass("checked")) {
179             labelIds += $(this).data("id") + ",";
180             $($ (this).data("id-selector")).removeClass("hide");
181         } else {
182             $($ (this).data("id-selector")).addClass("hide");
183         }
184     });
185     if (labelIds.length == 0) {
186         $noSelect.removeClass("hide");
187     } else {
188         $noSelect.addClass("hide");
189     }
190     $(
191         $(this)
192         .parent()
193         .data("id")
194     ).val(labelIds);
195     return false;
196 });
197 $labelMenu.find(".no-select.item").click(function() {
198     if (hasLabelUpdateAction) {
199         updateIssueMeta($labelMenu.data("update-url"), "clear", "");
200     }
201
202     $(this)
203     .parent()
204     .find(".item")
205     .each(function() {
206         $(this).removeClass("checked");
207         $(this)
208         .find(".octicon")
209         .removeClass("octicon-check")
210         .html("&nbsp;");
211     });
212
213     $list.find(".item").each(function() {
214         $(this).addClass("hide");
215     });
216     $noSelect.removeClass("hide");
217     $(
218         $(this)
219         .parent()
220         .data("id")
221     ).val("");
222 });
223
224 function selectItem(select_id, input_id) {

```

```

225     var $menu = $(select_id + " .menu");
226     var $list = $(".ui" + select_id + ".list");
227     var hasUpdateAction = $menu.data("action") == "update";
228
229     $menu.find(".item:not(.no-select)").click(function() {
230         $(this)
231             .parent()
232             .find(".item")
233             .each(function() {
234                 $(this).removeClass("selected active");
235             });
236
237         $(this).addClass("selected active");
238         if (hasUpdateAction) {
239             updateIssueMeta($menu.data("update-url"), "", $(this).data("id"));
240         }
241         switch (input_id) {
242             case "#milestone_id":
243                 $list
244                     .find(".selected")
245                     .html(
246                         '<a class="item" href=' +
247                             $(this).data("href") +
248                             ">" +
249                             $(this).text() +
250                             "</a>"
251                     );
252                 break;
253             case "#assignee_id":
254                 $list
255                     .find(".selected")
256                     .html(
257                         '<a class="item" href=' +
258                             $(this).data("href") +
259                             ">" +
260                             '<img class="ui avatar image" src=' +
261                             $(this).data("avatar") +
262                             ">" +
263                             $(this).text() +
264                             "</a>"
265                     );
266             }
267         $(".ui" + select_id + ".list .no-select").addClass("hide");
268         $(input_id).val($(this).data("id"));
269     });
270     $menu.find(".no-select.item").click(function() {
271         $(this)
272             .parent()
273             .find(".item:not(.no-select)")

```

```

274         .each(function() {
275             $(this).removeClass("selected active");
276         });
277
278         if (hasUpdateAction) {
279             updateIssueMeta($menu.data("update-url"), "", "");
280         }
281
282         $list.find(".selected").html("");
283         $list.find(".no-select").removeClass("hide");
284         $(input_id).val("");
285     });
286 }
287
288 // Milestone and assignee
289 selectItem(".select-milestone", "#milestone_id");
290 selectItem(".select-assignee", "#assignee_id");
291 }
292
293 function initRepository() {
294     if ($("#repository").length == 0) {
295         return;
296     }
297
298     function initFilterSearchDropdown(selector) {
299         var $dropdown = $(selector);
300         $dropdown.dropdown({
301             fullTextSearch: true,
302             onChange: function(text, value, $choice) {
303                 window.location.href = $choice.data("url");
304                 console.log($choice.data("url"));
305             },
306             message: { noResults: $dropdown.data("no-results") }
307         });
308     }
309
310     // File list and commits
311     if (
312         $("#repository.file.list").length > 0 ||
313         $("#repository.commits").length > 0
314     ) {
315         initFilterSearchDropdown(".choose.reference .dropdown");
316
317         $(".reference.column").click(function() {
318             $(".choose.reference .scrolling.menu").css("display", "none");
319             $(".choose.reference .text").removeClass("black");
320             $($(this).data("target")).css("display", "block");
321             $(this)
322                 .find(".text")

```

```

323         .addClass("black");
324     return false;
325 });
326 }
327
328 // Wiki
329 if ($("#repository.wiki.view").length > 0) {
330     initFilterSearchDropdown(".choose.page .dropdown");
331 }
332
333 // Options
334 if ($("#repository.settings.options").length > 0) {
335     $("#repo_name").keyup(function() {
336         var $prompt = $("#repo-name-change-prompt");
337         if (
338             $(this)
339                 .val()
340                 .toString()
341                 .toLowerCase() !=
342             $(this)
343                 .data("repo-name")
344                 .toString()
345                 .toLowerCase()
346         ) {
347             $prompt.show();
348         } else {
349             $prompt.hide();
350         }
351     });
352 }
353
354 // Branches
355 if ($("#repository.settings.branches").length > 0) {
356     initFilterSearchDropdown(".protected-branches .dropdown");
357     $(".enable-protection, .enable-whitelist").change(function() {
358         if (this.checked) {
359             $($ (this).data("target")).removeClass("disabled");
360         } else {
361             $($ (this).data("target")).addClass("disabled");
362         }
363     });
364 }
365
366 // Labels
367 if ($("#repository.labels").length > 0) {
368     // Create label
369     var $newLabelPanel = $(".new-label.segment");
370     $(".new-label.button").click(function() {
371         $newLabelPanel.show();

```



```

372     });
373     $(".new-label.segment .cancel").click(function() {
374         $newLabelPanel.hide();
375     });
376
377     $(".color-picker").each(function() {
378         $(this).minicolors();
379     });
380     $(".precolors .color").click(function() {
381         var color_hex = $(this).data("color-hex");
382         $(".color-picker").val(color_hex);
383         $(".minicolors-swatch-color").css("background-color", color_hex);
384     });
385     $(".edit-label-button").click(function() {
386         $("#label-modal-id").val($(this).data("id"));
387         $(".edit-label .new-label-input").val($(this).data("title"));
388         $(".edit-label .color-picker").val($(this).data("color"));
389         $(".minicolors-swatch-color").css(
390             "background-color",
391             $(this).data("color")
392         );
393         $(".edit-label.modal")
394             .modal({
395                 onApprove: function() {
396                     $(".edit-label.form").submit();
397                 }
398             })
399             .modal("show");
400         return false;
401     });
402 }
403
404 // Milestones
405 if ($("#repository.milestones").length > 0) {
406 }
407 if ($("#repository.new.milestone").length > 0) {
408     var $datepicker = $(".milestone.datepicker");
409     $datepicker.datetimepicker({
410         lang: $datepicker.data("lang"),
411         inline: true,
412         timepicker: false,
413         startDate: $datepicker.data("start-date"),
414         formatDate: "Y-m-d",
415         onSelectDate: function(ct) {
416             $("#deadline").val(ct.dateFormat("Y-m-d"));
417         }
418     });
419     $("#clear-date").click(function() {
420         $("#deadline").val("");

```

```
421     return false;
422   });
423 }
424
425 // Issues
426 if ($("#repository.view.issue").length > 0) {
427   // Edit issue title
428   var $issueTitle = $("#issue-title");
429   var $editInput = $("#edit-title-input").find("input");
430   var editTitleToggle = function() {
431     $issueTitle.toggle();
432     $(".not-in-edit").toggle();
433     $("#edit-title-input").toggle();
434     $(".in-edit").toggle();
435     $editInput.focus();
436     return false;
437   };
438   $("#edit-title").click(editTitleToggle);
439   $("#cancel-edit-title").click(editTitleToggle);
440   $("#save-edit-title")
441     .click(editTitleToggle)
442     .click(function() {
443       if (
444         $editInput.val().length == 0 ||
445         $editInput.val() == $issueTitle.text()
446       ) {
447         $editInput.val($issueTitle.text());
448         return false;
449       }
450
451       $.post(
452         $(this).data("update-url"),
453         {
454           _csrf: csrf,
455           title: $editInput.val()
456         },
457         function(data) {
458           $editInput.val(data.title);
459           $issueTitle.text(data.title);
460         }
461       );
462       return false;
463     });
464
465   // Edit issue or comment content
466   $(".edit-content").click(function() {
467     var $segment = $(this)
468       .parent()
469       .parent()
```

```

470     .parent()
471     .next();
472     var $editContentZone = $segment.find(".edit-content-zone");
473     var $renderContent = $segment.find(".render-content");
474     var $rawContent = $segment.find(".raw-content");
475     var $textarea;
476
477     // Setup new form
478     if ($editContentZone.html().length == 0) {
479         $editContentZone.html($("#edit-content-form").html());
480         $textarea = $segment.find("textarea");
481
482         // Give new write/preview data-tab name to distinguish from others
483         var $editContentForm = $editContentZone.find(".ui.comment.form");
484         var $tabMenu = $editContentForm.find(".tabular.menu");
485         $tabMenu.attr("data-write", $editContentZone.data("write"));
486         $tabMenu.attr("data-preview", $editContentZone.data("preview"));
487         $tabMenu
488             .find(".write.item")
489             .attr("data-tab", $editContentZone.data("write"));
490         $tabMenu
491             .find(".preview.item")
492             .attr("data-tab", $editContentZone.data("preview"));
493         $editContentForm
494             .find(".write.segment")
495             .attr("data-tab", $editContentZone.data("write"));
496         $editContentForm
497             .find(".preview.segment")
498             .attr("data-tab", $editContentZone.data("preview"));
499
500         initCommentPreviewTab($editContentForm);
501
502         $editContentZone.find(".cancel.button").click(function() {
503             $renderContent.show();
504             $editContentZone.hide();
505         });
506         $editContentZone.find(".save.button").click(function() {
507             $renderContent.show();
508             $editContentZone.hide();
509
510             $.post(
511                 $editContentZone.data("update-url"),
512                 {
513                     _csrf: csrf,
514                     content: $textarea.val(),
515                     context: $editContentZone.data("context")
516                 },
517                 function(data) {
518                     if (data.length == 0) {

```

```

519         $renderContent.html($("#no-content").html());
520     } else {
521         $renderContent.html(data.content);
522         emojiify.run($renderContent[0]);
523         $("pre code", $renderContent[0]).each(function(i, block) {
524             hljs.highlightBlock(block);
525         });
526     }
527 }
528 );
529 });
530 } else {
531     $textarea = $segment.find("textarea");
532 }
533
534 // Show write/preview tab and copy raw content as needed
535 $editContentZone.show();
536 $renderContent.hide();
537 if ($textarea.val().length == 0) {
538     $textarea.val($rawContent.text());
539 }
540 $textarea.focus();
541 return false;
542 });
543
544 // Delete comment
545 $(".delete-comment").click(function() {
546     var $this = $(this);
547     if (confirm($this.data("locale"))) {
548         $.post($this.data("url"), {
549             _csrf: csrf
550         }).done(function() {
551             $("##" + $this.data("comment-id")).remove();
552         });
553     }
554     return false;
555 });
556
557 // Change status
558 var $statusButton = $("#status-button");
559 $("#comment-form .edit_area").keyup(function() {
560     if ($(this).val().length == 0) {
561         $statusButton.text($statusButton.data("status"));
562     } else {
563         $statusButton.text($statusButton.data("status-and-comment"));
564     }
565 });
566 $statusButton.click(function() {
567     $("#status").val($statusButton.data("status-val"));

```

```

568     $("#comment-form").submit();
569 });
570 }
571
572 // Diff
573 if ($.repository.diff).length > 0) {
574     var $counter = $(".diff-counter");
575     if ($counter.length >= 1) {
576         $counter.each(function(i, item) {
577             var $item = $(item);
578             var addLine = $item.find("span[data-line].add").data("line");
579             var delLine = $item.find("span[data-line].del").data("line");
580             var addPercent =
581                 (parseFloat(addLine) / (parseFloat(addLine) + parseFloat(delLine))) *
582                 100;
583             $item.find(".bar .add").css("width", addPercent + "%");
584         });
585     }
586
587     $(".diff-file-box .lines-num").click(function() {
588         if ($(this).attr("id")) {
589             window.location.href = "#" + $(this).attr("id");
590         }
591     });
592
593     $(window)
594         .on("hashchange", function(e) {
595             $(".diff-file-box .lines-code.active").removeClass("active");
596             var m = window.location.hash.match(/^#diff-.+$/);
597             if (m) {
598                 $(m[0])
599                     .siblings(".lines-code")
600                     .addClass("active");
601             }
602         })
603         .trigger("hashchange");
604 }
605
606 // Quick start and repository home
607 $("#repo-clone-ssh").click(function() {
608     $(".clone-url").text($(this).data("link"));
609     $("#repo-clone-url").val($(this).data("link"));
610     $(this).addClass("blue");
611     $("#repo-clone-https").removeClass("blue");
612     localStorage.setItem("repo-clone-protocol", "ssh");
613 });
614 $("#repo-clone-https").click(function() {
615     $(".clone-url").text($(this).data("link"));
616     $("#repo-clone-url").val($(this).data("link"));

```

```

617     $(this).addClass("blue");
618     $("#repo-clone-ssh").removeClass("blue");
619     localStorage.setItem("repo-clone-protocol", "https");
620 });
621 $("#repo-clone-url").click(function() {
622     $(this).select();
623 });
624
625 // Pull request
626 if ($("#repository.compare.pull").length > 0) {
627     initFilterSearchDropdown(".choose.branch .dropdown");
628 }
629 if ($("#repository.view.pull").length > 0) {
630     $(".comment.merge.box input[name=merge_style]").change(function() {
631         if ($(this).val() === "create_merge_commit") {
632             $(".commit.description.field").show();
633         } else {
634             $(".commit.description.field").hide();
635         }
636     });
637 }
638 }
639
640 function initWikiForm() {
641     var $editArea = $("#repository.wiki textarea#edit_area");
642     if ($editArea.length > 0) {
643         new SimpleMDE({
644             autoDownloadFontAwesome: false,
645             element: $editArea[0],
646             forceSync: true,
647             previewRender: function(plainText, preview) {
648                 // Async method
649                 setTimeout(function() {
650                     // FIXME: still send render request when return back to edit mode
651                     $.post(
652                         $editArea.data("url"),
653                         {
654                             _csrf: csrf,
655                             mode: "gfm",
656                             context: $editArea.data("context"),
657                             text: plainText
658                         },
659                         function(data) {
660                             preview.innerHTML = '<div class="markdown">' + data + "</div>";
661                             emojiify.run($(".editor-preview")[0]);
662                         }
663                     );
664                 }, 0);
665             }

```

```

666         return "Loading...";
667     },
668     renderingConfig: {
669         singleLineBreaks: false
670     },
671     indentWithTabs: false,
672     tabSize: 4,
673     spellChecker: false,
674     toolbar: [
675         "bold",
676         "italic",
677         "strikethrough",
678         "|",
679         "heading-1",
680         "heading-2",
681         "heading-3",
682         "heading-bigger",
683         "heading-smaller",
684         "|",
685         "code",
686         "quote",
687         "|",
688         "unordered-list",
689         "ordered-list",
690         "|",
691         "link",
692         "image",
693         "table",
694         "horizontal-rule",
695         "|",
696         "clean-block",
697         "preview",
698         "fullscreen"
699     ]
700     });
701 }
702 }
703
704 var simpleMDEditor;
705 var codeMirrorEditor;
706
707 // For IE
708 String.prototype.endsWith = function(pattern) {
709     var d = this.length - pattern.length;
710     return d >= 0 && this.lastIndexOf(pattern) === d;
711 };
712
713 // Adding function to get the cursor position in a text field to jQuery object.
714 (function($, undefined) {

```

```

715     $.fn.setCursorPosition = function() {
716         var el = $(this).get(0);
717         var pos = 0;
718         if ("selectionStart" in el) {
719             pos = el.selectionStart;
720         } else if ("selection" in document) {
721             el.focus();
722             var Sel = document.selection.createRange();
723             var Sellength = document.selection.createRange().text.length;
724             Sel.moveStart("character", -el.value.length);
725             pos = Sel.text.length - Sellength;
726         }
727         return pos;
728     };
729 })(jQuery);
730
731 function setSimpleMDE($editArea) {
732     if (codeMirrorEditor) {
733         codeMirrorEditor.toTextArea();
734         codeMirrorEditor = null;
735     }
736
737     if (simpleMDEditor) {
738         return true;
739     }
740
741     simpleMDEditor = new SimpleMDE({
742         autoDownloadFontAwesome: false,
743         element: $editArea[0],
744         forceSync: true,
745         renderingConfig: {
746             singleLineBreaks: false
747         },
748         indentWithTabs: false,
749         tabSize: 4,
750         spellChecker: false,
751         previewRender: function(plainText, preview) {
752             // Async method
753             setTimeout(function() {
754                 // FIXME: still send render request when return back to edit mode
755                 $.post(
756                     $editArea.data("url"),
757                     {
758                         _csrf: csrf,
759                         mode: "gfm",
760                         context: $editArea.data("context"),
761                         text: plainText
762                     },
763                     function(data) {

```



```

764         preview.innerHTML = '<div class="markdown">' + data + "</div>";
765         emojiify.run($(".editor-preview")[0]);
766     }
767     );
768     }, 0);
769
770     return "Loading...";
771 },
772 toolbar: [
773     "bold",
774     "italic",
775     "strikethrough",
776     "|",
777     "heading-1",
778     "heading-2",
779     "heading-3",
780     "heading-bigger",
781     "heading-smaller",
782     "|",
783     "code",
784     "quote",
785     "|",
786     "unordered-list",
787     "ordered-list",
788     "|",
789     "link",
790     "image",
791     "table",
792     "horizontal-rule",
793     "|",
794     "clean-block"
795 ]
796 });
797
798 return true;
799 }
800
801 function setCodeMirror($editArea) {
802     if (simpleMDEditor) {
803         simpleMDEditor.toTextArea();
804         simpleMDEditor = null;
805     }
806
807     if (codeMirrorEditor) {
808         return true;
809     }
810
811     codeMirrorEditor = CodeMirror.fromTextArea($editArea[0], {
812         lineNumbers: true

```

```

813 });
814 codeMirrorEditor.on("change", function(cm, change) {
815     $editArea.val(cm.getValue());
816 });
817
818 return true;
819 }
820
821 function initEditor() {
822     $(".js-quick-pull-choice-option").change(function() {
823         if ($(this).val() == "commit-to-new-branch") {
824             $(".quick-pull-branch-name").show();
825             $(".quick-pull-branch-name input").prop("required", true);
826         } else {
827             $(".quick-pull-branch-name").hide();
828             $(".quick-pull-branch-name input").prop("required", false);
829         }
830     });
831
832     var $editFilename = $("#file-name");
833     $editFilename
834         .keyup(function(e) {
835             var $section = $(".breadcrumb span.section");
836             var $divider = $(".breadcrumb div.divider");
837             if (e.keyCode == 8) {
838                 if ($(this).getCursorPosition() == 0) {
839                     if ($section.length > 0) {
840                         var value = $section
841                             .last()
842                             .find("a")
843                             .text();
844                         $(this).val(value + $(this).val());
845                         $(this)[0].setSelectionRange(value.length, value.length);
846                         $section.last().remove();
847                         $divider.last().remove();
848                     }
849                 }
850             }
851             if (e.keyCode == 191) {
852                 var parts = $(this)
853                     .val()
854                     .split("/");
855                 for (var i = 0; i < parts.length; ++i) {
856                     var value = parts[i];
857                     if (i < parts.length - 1) {
858                         if (value.length) {
859                             $(
860                                 '<span class="section"><a href="#">' + value + "</a></span>'
861                             ).insertBefore($(this));

```

```

862         $('<div class="divider"> / </div>').insertBefore($(this));
863     }
864 } else {
865     $(this).val(value);
866 }
867 $(this)[0].setSelectionRange(0, 0);
868 }
869 }
870 var parts = [];
871 $(".breadcrumb span.section").each(function(i, element) {
872     element = $(element);
873     if (element.find("a").length) {
874         parts.push(element.find("a").text());
875     } else {
876         parts.push(element.text());
877     }
878 });
879 if ($(this).val()) {
880     parts.push($(this).val());
881 }
882
883 var tree_path = parts.join("/");
884 $("#tree_path").val(tree_path);
885 $("#preview-tab").data(
886     "context",
887     $("#preview-tab").data("root-context") +
888     tree_path.substring(0, tree_path.lastIndexOf("/") + 1)
889 );
890 })
891 .trigger("keyup");
892
893 var $editArea = $(".repository.editor textarea#edit_area");
894 if (!$editArea.length) return;
895
896 var markdownFileExts = $editArea.data("markdown-file-exts").split(",");
897 var lineWrapExtensions = $editArea.data("line-wrap-extensions").split(",");
898
899 $editFilename
900 .on("keyup", function(e) {
901     var val = $editFilename.val(),
902         m,
903         mode,
904         spec,
905         extension,
906         extWithDot,
907         previewLink,
908         dataUrl,
909         apiCall;
910     extension = extWithDot = "";

```

```
911     if ((m = /.+\.[^.]+$/).exec(val)) {
912         extension = m[1].toLowerCase();
913         extWithDot = "." + extension;
914     }
915
916     var info = CodeMirror.findModeByExtension(extension);
917     previewLink = $("a[data-tab=preview]");
918     if (info) {
919         mode = info.mode;
920         spec = info.mime;
921         apiCall = mode;
922     } else {
923         apiCall = extension;
924     }
925
926     if (
927         previewLink.length &&
928         apiCall &&
929         previewFileModes &&
930         previewFileModes.length &&
931         previewFileModes.indexOf(apiCall) >= 0
932     ) {
933         dataUrl = previewLink.data("url");
934         previewLink.data("url", dataUrl.replace(/(.*)\./, "$1/" + mode));
935         previewLink.show();
936     } else {
937         previewLink.hide();
938     }
939
940     // If this file is a Markdown extensions, we will load that editor and return
941     if (markdownFileExts.indexOf(extWithDot) >= 0) {
942         if (setSimpleMDE($editArea)) {
943             return;
944         }
945     }
946
947     // Else we are going to use CodeMirror
948     if (!codeMirrorEditor && !setCodeMirror($editArea)) {
949         return;
950     }
951
952     if (mode) {
953         codeMirrorEditor.setOption("mode", spec);
954         CodeMirror.autoLoadMode(codeMirrorEditor, mode);
955     }
956
957     if (lineWrapExtensions.indexOf(extWithDot) >= 0) {
958         codeMirrorEditor.setOption("lineWrapping", true);
959     } else {
```

```

960     codeMirrorEditor.setOption("lineWrapping", false);
961 }
962
963 // get the filename without any folder
964 var value = $editFilename.val();
965 if (value.length === 0) {
966     return;
967 }
968 value = value.split("/");
969 value = value[value.length - 1];
970
971 $.getJSON($editFilename.data("ec-url-prefix") + value, function(
972     editorconfig
973 ) {
974     if (editorconfig.indent_style === "tab") {
975         codeMirrorEditor.setOption("indentWithTabs", true);
976         codeMirrorEditor.setOption("extraKeys", {});
977     } else {
978         codeMirrorEditor.setOption("indentWithTabs", false);
979         // required because CodeMirror doesn't seems to use spaces correctly for {"indentWithTab
980         // - https://github.com/codemirror/CodeMirror/issues/988
981         // - https://codemirror.net/doc/manual.html#keymaps
982         codeMirrorEditor.setOption("extraKeys", {
983             Tab: function(cm) {
984                 var spaces = Array(parseInt(cm.getOption("indentUnit")) + 1).join(
985                     " "
986                 );
987                 cm.replaceSelection(spaces);
988             }
989         });
990     }
991     codeMirrorEditor.setOption("indentUnit", editorconfig.indent_size || 4);
992     codeMirrorEditor.setOption("tabSize", editorconfig.tab_width || 4);
993 });
994 })
995 .trigger("keyup");
996 }
997
998 function initOrganization() {
999     if ($("#organization").length == 0) {
1000         return;
1001     }
1002
1003     // Options
1004     if ($("#organization.settings.options").length > 0) {
1005         $("#org_name").keyup(function() {
1006             var $prompt = $("#org-name-change-prompt");
1007             if (
1008                 $(this)

```

```
1009         .val()
1010         .toString()
1011         .toLowerCase() !=
1012         $(this)
1013         .data("org-name")
1014         .toString()
1015         .toLowerCase()
1016     ) {
1017         $prompt.show();
1018     } else {
1019         $prompt.hide();
1020     }
1021 });
1022 }
1023 }
1024
1025 function initAdmin() {
1026     if ($("#admin").length == 0) {
1027         return;
1028     }
1029
1030     // New user
1031     if ($("#admin.new.user").length > 0 || ($("#admin.edit.user").length > 0) {
1032         $("#login_type").change(function() {
1033             if (
1034                 $(this)
1035                 .val()
1036                 .substring(0, 1) == "0"
1037             ) {
1038                 $("#login_name").removeAttr("required");
1039                 $(".non-local").hide();
1040                 $(".local").show();
1041                 $("#user_name").focus();
1042
1043                 if ($(this).data("password") == "required") {
1044                     $("#password").attr("required", "required");
1045                 }
1046             } else {
1047                 $("#login_name").attr("required", "required");
1048                 $(".non-local").show();
1049                 $(".local").hide();
1050                 $("#login_name").focus();
1051
1052                 $("#password").removeAttr("required");
1053             }
1054         });
1055     }
1056
1057     function onSecurityProtocolChange() {
```

```
1058     if ($("#security_protocol").val() > 0) {
1059         $(".has-tls").show();
1060     } else {
1061         $(".has-tls").hide();
1062     }
1063 }
1064
1065 // New authentication
1066 if ($("#admin.new.authentication").length > 0) {
1067     $("#auth_type").change(function() {
1068         $(".ldap").hide();
1069         $(".dldap").hide();
1070         $(".smtp").hide();
1071         $(".pam").hide();
1072         $(".github").hide();
1073         $(".has-tls").hide();
1074
1075         var authType = $(this).val();
1076         switch (authType) {
1077             case "2": // LDAP
1078                 $(".ldap").show();
1079                 break;
1080             case "3": // SMTP
1081                 $(".smtp").show();
1082                 $(".has-tls").show();
1083                 break;
1084             case "4": // PAM
1085                 $(".pam").show();
1086                 break;
1087             case "5": // LDAP
1088                 $(".dldap").show();
1089                 break;
1090             case "6": //GITHUB
1091                 $(".github").show();
1092                 break;
1093         }
1094
1095         if (authType == "2" || authType == "5") {
1096             onSecurityProtocolChange();
1097         }
1098     });
1099     $("#security_protocol").change(onSecurityProtocolChange);
1100 }
1101 // Edit authentication
1102 if ($("#admin.edit.authentication").length > 0) {
1103     var authType = $("#auth_type").val();
1104     if (authType == "2" || authType == "5") {
1105         $("#security_protocol").change(onSecurityProtocolChange);
1106     }
```

```
1107     }
1108
1109     // Notice
1110     if ($("#admin.notice")) {
1111         var $detailModal = $("#detail-modal");
1112
1113         // Attach view detail modals
1114         $(".view-detail").click(function() {
1115             $detailModal.find(".content p").text($(this).data("content"));
1116             $detailModal.modal("show");
1117             return false;
1118         });
1119
1120         // Select actions
1121         var $checkboxes = $(".select.table .ui.checkbox");
1122         $(".select.action").click(function() {
1123             switch ($(this).data("action")) {
1124                 case "select-all":
1125                     $checkboxes.checkbox("check");
1126                     break;
1127                 case "deselect-all":
1128                     $checkboxes.checkbox("uncheck");
1129                     break;
1130                 case "inverse":
1131                     $checkboxes.checkbox("toggle");
1132                     break;
1133             }
1134         });
1135         $("#delete-selection").click(function() {
1136             var $this = $(this);
1137             $this.addClass("loading disabled");
1138             var ids = [];
1139             $checkboxes.each(function() {
1140                 if ($(this).checkbox("is checked")) {
1141                     ids.push($(this).data("id"));
1142                 }
1143             });
1144             $.post($this.data("link"), {
1145                 _csrf: csrf,
1146                 ids: ids
1147             }).done(function() {
1148                 window.location.href = $this.data("redirect");
1149             });
1150         });
1151     }
1152 }
1153
1154 function buttonsClickOnEnter() {
1155     $(".ui.button").keypress(function(e) {
```



```

1156     if (e.keyCode == 13 || e.keyCode == 32)
1157         // enter key or space bar
1158         $(this).click();
1159     });
1160 }
1161
1162 function hideWhenLostFocus(body, parent) {
1163     $(document).click(function(e) {
1164         var target = e.target;
1165         if (
1166             !$(target).is(body) &&
1167             !$(target)
1168                 .parents()
1169                 .is(parent)
1170         ) {
1171             $(body).hide();
1172         }
1173     });
1174 }
1175
1176 function searchUsers() {
1177     if ($("#search-user-box .results").length) {
1178         return;
1179     }
1180
1181     var $searchUserBox = $("#search-user-box");
1182     var $results = $searchUserBox.find(".results");
1183     $searchUserBox.keyup(function() {
1184         var $this = $(this);
1185         var keyword = $this.find("input").val();
1186         if (keyword.length < 2) {
1187             $results.hide();
1188             return;
1189         }
1190
1191         $.ajax({
1192             url: suburl + "/api/v1/users/search?q=" + keyword,
1193             dataType: "json",
1194             success: function(response) {
1195                 var notEmpty = function(str) {
1196                     return str && str.length > 0;
1197                 };
1198
1199                 $results.html("");
1200
1201                 if (response.ok && response.data.length) {
1202                     var html = "";
1203                     $.each(response.data, function(i, item) {
1204                         html +=

```

```

1205         '<div class="item"><span class="username">' +
1208         item.username +
1209         "</span>";
1210         if (notEmpty(item.full_name)) {
1211             html += " (" + item.full_name + ")";
1212         }
1213         html += "</div>";
1214     });
1215     $results.html(html);
1216     $this.find(".results .item").click(function() {
1217         $this.find("input").val(
1218             $(this)
1219                 .find(".username")
1220                 .text()
1221         );
1222         $results.hide();
1223     });
1224     $results.show();
1225 } else {
1226     $results.hide();
1227 }
1228 }
1229 });
1230 });
1231 $searchUserBox.find("input").focus(function() {
1232     $searchUserBox.keyup();
1233 });
1234 hideWhenLostFocus("#search-user-box .results", "#search-user-box");
1235 }
1236
1237 // FIXME: merge common parts in two functions
1238 function searchRepositories() {
1239     if (!$("#search-repo-box .results").length) {
1240         return;
1241     }
1242
1243     var $searchRepoBox = $("#search-repo-box");
1244     var $results = $searchRepoBox.find(".results");
1245     $searchRepoBox.keyup(function() {
1246         var $this = $(this);
1247         var keyword = $this.find("input").val();
1248         if (keyword.length < 2) {
1249             $results.hide();
1250             return;
1251         }
1252
1253         $.ajax({

```

```

1254     url:
1255         suburl +
1256         "/api/v1/repos/search?q=" +
1257         keyword +
1258         "&uid=" +
1259         $searchRepoBox.data("uid"),
1260     dataType: "json",
1261     success: function(response) {
1262         var notEmpty = function(str) {
1263             return str && str.length > 0;
1264         };
1265
1266         $results.html("");
1267
1268         if (response.ok && response.data.length) {
1269             var html = "";
1270             $.each(response.data, function(i, item) {
1271                 html +=
1272                     '<div class="item"><i class="octicon octicon-repo"></i> <span class="fullname">' +
1273                     item.full_name +
1274                     "</span></div>";
1275             });
1276             $results.html(html);
1277             $this.find(".results .item").click(function() {
1278                 $this.find("input").val(
1279                     $(this)
1280                         .find(".fullname")
1281                         .text()
1282                         .split("/")[1]
1283                 );
1284                 $results.hide();
1285             });
1286             $results.show();
1287         } else {
1288             $results.hide();
1289         }
1290     }
1291 });
1292 });
1293 $searchRepoBox.find("input").focus(function() {
1294     $searchRepoBox.keyup();
1295 });
1296 hideWhenLostFocus("#search-repo-box .results", "#search-repo-box");
1297 }
1298
1299 function initCodeView() {
1300     if ($("#code-view .linenums").length > 0) {
1301         $(document).on("click", ".lines-num span", function(e) {
1302             var $select = $(this);

```

```

1303     var $list = $select
1304         .parent()
1305         .siblings(".lines-code")
1306         .find("ol.linenums > li");
1307     selectRange(
1308         $list,
1309         $list.filter("[rel=" + $select.attr("id") + "]"),
1310         e.shiftKey ? $list.filter(".active").eq(0) : null
1311     );
1312     deSelect();
1313 });
1314
1315 $(window)
1316     .on("hashchange", function(e) {
1317         var m = window.location.hash.match(/^#(L\d+)\-(L\d+)$/);
1318         var $list = $(".code-view ol.linenums > li");
1319         var $first;
1320         if (m) {
1321             $first = $list.filter("." + m[1]);
1322             selectRange($list, $first, $list.filter("." + m[2]));
1323             $("html, body").scrollTop($first.offset().top - 200);
1324             return;
1325         }
1326         m = window.location.hash.match(/^#(L\d+)$/);
1327         if (m) {
1328             $first = $list.filter("." + m[1]);
1329             selectRange($list, $first);
1330             $("html, body").scrollTop($first.offset().top - 200);
1331         }
1332     })
1333     .trigger("hashchange");
1334 }
1335 }
1336
1337 function initUserSettings() {
1338     console.log("initUserSettings");
1339
1340     // Options
1341     if ($("#user.settings.profile").length > 0) {
1342         $("#username").keyup(function() {
1343             var $prompt = $("#name-change-prompt");
1344             if (
1345                 $(this)
1346                     .val()
1347                     .toString()
1348                     .toLowerCase() !=
1349                 $(this)
1350                     .data("name")
1351                     .toString()

```

```

1352         .toLowerCase()
1353     ) {
1354         $prompt.show();
1355     } else {
1356         $prompt.hide();
1357     }
1358 });
1359 }
1360 }
1361
1362 function initRepositoryCollaboration() {
1363     console.log("initRepositoryCollaboration");
1364
1365     // Change collaborator access mode
1366     $(".access-mode.menu .item").click(function() {
1367         var $menu = $(this).parent();
1368         $.post($menu.data("url"), {
1369             _csrf: csrf,
1370             uid: $menu.data("uid"),
1371             mode: $(this).data("value")
1372         });
1373     });
1374 }
1375
1376 function initWebhookSettings() {
1377     $(".events.checkbox input").change(function() {
1378         if ($(this).is(":checked")) {
1379             $(".events.fields").show();
1380         }
1381     });
1382     $(".non-events.checkbox input").change(function() {
1383         if ($(this).is(":checked")) {
1384             $(".events.fields").hide();
1385         }
1386     });
1387
1388     // Highlight payload on first click
1389     $(".hook.history.list .toggle.button").click(function() {
1390         $($($(this).data("target") + " .nohighlight").each(function() {
1391             var $this = $(this);
1392             $this.removeClass("nohighlight");
1393             setTimeout(function() {
1394                 hljs.highlightBlock($this[0]);
1395             }, 500);
1396         }));
1397     });
1398
1399     // Trigger delivery
1400     $(".delivery.button, .redelivery.button").click(function() {

```

```
1401     var $this = $(this);
1402     $this.addClass("loading disabled");
1403     $.post($this.data("link"), {
1404         _csrf: csrf
1405     }).done(
1406         setTimeout(function() {
1407             window.location.href = $this.data("redirect");
1408         }, 5000)
1409     );
1410 });
1411 }
1412
1413 $(document).ready(function() {
1414     csrf = $("meta[name=_csrf]").attr("content");
1415     suburl = $("meta[name=_suburl]").attr("content");
1416
1417     // Set cursor to the end of autofocus input string
1418     $("input[autofocus]").each(function() {
1419         $(this).val($(this).val());
1420     });
1421
1422     // Show exact time
1423     $(".time-since").each(function() {
1424         $(this)
1425             .addClass("popping up")
1426             .attr("data-content", $(this).attr("title"))
1427             .attr("data-variation", "inverted tiny")
1428             .attr("title", "");
1429     });
1430
1431     // Semantic UI modules.
1432     $(".ui.dropdown").dropdown({
1433         forceSelection: false
1434     });
1435     $(".jump.dropdown").dropdown({
1436         action: "select",
1437         onShow: function() {
1438             $(".popping.up").popup("hide");
1439         }
1440     });
1441     $(".slide.up.dropdown").dropdown({
1442         transition: "slide up"
1443     });
1444     $(".upward.dropdown").dropdown({
1445         direction: "upward"
1446     });
1447     $(".ui.accordion").accordion();
1448     $(".ui.checkbox").checkbox();
1449     $(".ui.progress").progress({
```

```

1450     showActivity: false
1451   });
1452   $(".popping.up").popup();
1453   $(".top.menu .popping.up").popup({
1454     onShow: function() {
1455       if $(".top.menu .menu.transition").hasClass("visible") {
1456         return false;
1457       }
1458     }
1459   });
1460   $(".tabular.menu .item").tab();
1461   $(".tabable.menu .item").tab();
1462
1463   $(".toggle.button").click(function() {
1464     $(".this").data("target").slideToggle(100);
1465   });
1466
1467   // Dropzone
1468   var $dropzone = $("#dropzone");
1469   if ($dropzone.length > 0) {
1470     var filenameDict = {};
1471     $dropzone.dropzone({
1472       url: $dropzone.data("upload-url"),
1473       headers: { "X-CSRF-Token": csrf },
1474       maxFiles: $dropzone.data("max-file"),
1475       maxFileSize: $dropzone.data("max-size"),
1476       acceptedFiles:
1477         $dropzone.data("accepts") === "*/*" ? null : $dropzone.data("accepts"),
1478       addRemoveLinks: true,
1479       dictDefaultMessage: $dropzone.data("default-message"),
1480       dictInvalidFileType: $dropzone.data("invalid-input-type"),
1481       dictFileTooBig: $dropzone.data("file-too-big"),
1482       dictRemoveFile: $dropzone.data("remove-file"),
1483       init: function() {
1484         this.on("success", function(file, data) {
1485           filenameDict[file.name] = data.uuid;
1486           var input = $(
1487             '<input id="' + data.uuid + '" name="files" type="hidden">'
1488           ).val(data.uuid);
1489           $(".files").append(input);
1490         });
1491         this.on("removedfile", function(file) {
1492           if (file.name in filenameDict) {
1493             $(".#" + filenameDict[file.name]).remove();
1494           }
1495           if ($dropzone.data("remove-url") && $dropzone.data("csrf")) {
1496             $.post($dropzone.data("remove-url"), {
1497               file: filenameDict[file.name],
1498               csrf: $dropzone.data("csrf")

```

```
1499         });
1500     }
1501     });
1502 }
1503 });
1504 }
1505
1506 // Emojify
1507 emojify.setConfig({
1508     img_dir: suburl + "/img/emoji",
1509     ignore_emoticons: true
1510 });
1511 var hasEmoji = document.getElementsByClassName("has-emoji");
1512 for (var i = 0; i < hasEmoji.length; i++) {
1513     emojify.run(hasEmoji[i]);
1514 }
1515
1516 // Clipboard JS
1517 var clipboard = new ClipboardJS(".clipboard");
1518 clipboard.on("success", function(e) {
1519     e.clearSelection();
1520
1521     $("# + e.trigger.getAttribute("id")).popup("destroy");
1522     e.trigger.setAttribute(
1523         "data-content",
1524         e.trigger.getAttribute("data-success")
1525     );
1526     $("# + e.trigger.getAttribute("id")).popup("show");
1527     e.trigger.setAttribute(
1528         "data-content",
1529         e.trigger.getAttribute("data-original")
1530     );
1531 });
1532
1533 clipboard.on("error", function(e) {
1534     $("# + e.trigger.getAttribute("id")).popup("destroy");
1535     e.trigger.setAttribute(
1536         "data-content",
1537         e.trigger.getAttribute("data-error")
1538     );
1539     $("# + e.trigger.getAttribute("id")).popup("show");
1540     e.trigger.setAttribute(
1541         "data-content",
1542         e.trigger.getAttribute("data-original")
1543     );
1544 });
1545
1546 // Autosize
1547 if ($("#description.autosize").length > 0) {
```



```

1548     autosize($("#description"));
1549     showMessageMaxLength(512, "description", "descLength");
1550 }
1551
1552 // AJAX load buttons
1553 $(".ajax-load-button").click(function() {
1554     var $this = $(this);
1555     $this.addClass("disabled");
1556
1557     $.ajax({
1558         url: $this.data("url"),
1559         headers: {
1560             "X-AJAX": "true"
1561         }
1562     }).done(function(data, status, request) {
1563         $(data).insertBefore($this);
1564
1565         // Update new URL or remove self if no more feeds
1566         var url = request.getResponseHeader("X-AJAX-URL");
1567         if (url) {
1568             $this.data("url", url);
1569             $this.removeClass("disabled");
1570         } else {
1571             $this.remove();
1572         }
1573     });
1574 });
1575
1576 // Helpers
1577 $(".delete-button").click(function() {
1578     var $this = $(this);
1579     $(".delete.modal")
1580         .modal({
1581             closable: false,
1582             onApprove: function() {
1583                 if ($this.data("type") == "form") {
1584                     $($this.data("form")).submit();
1585                     return;
1586                 }
1587
1588                 $.post($this.data("url"), {
1589                     _csrf: csrf,
1590                     id: $this.data("id")
1591                 }).done(function(data) {
1592                     window.location.href = data.redirect;
1593                 });
1594             }
1595         })
1596         .modal("show");

```

```

1597     return false;
1598 });
1599 $(".show-panel.button").click(function() {
1600     $($ (this).data("panel")).show();
1601 });
1602 $(".show-modal.button").click(function() {
1603     $($ (this).data("modal")).modal("show");
1604 });
1605 $(".delete-post.button").click(function() {
1606     var $this = $(this);
1607     $.post($this.data("request-url"), {
1608         _csrf: csrf
1609     }).done(function() {
1610         window.location.href = $this.data("done-url");
1611     });
1612 });
1613 // To make arbitrary form element to behave like a submit button
1614 $(".submit-button").click(function() {
1615     $($ (this).data("form")).submit();
1616 });
1617
1618 // Check or select on option to enable/disable target region
1619 $(".enable-system").change(function() {
1620     if (this.checked) {
1621         $($ (this).data("target")).removeClass("disabled");
1622     } else {
1623         $($ (this).data("target")).addClass("disabled");
1624         $($ (this).data("uncheck")).prop("checked", false);
1625     }
1626 });
1627 $(".enable-system-radio").change(function() {
1628     $($ (this).data("enable")).removeClass("disabled");
1629     $($ (this).data("disable")).addClass("disabled");
1630     $($ (this).data("uncheck")).prop("checked", false);
1631 });
1632
1633 // Set anchor.
1634 $(".markdown").each(function() {
1635     var headers = {};
1636     $(this)
1637         .find("h1, h2, h3, h4, h5, h6")
1638         .each(function() {
1639             var node = $(this);
1640             var val = encodeURIComponent(
1641                 node
1642                     .text()
1643                     .toLowerCase()
1644                     .replace(/[\^\\u00C0-\u1FFF\u2C00-\uD7FFw\ - ]/g, "")
1645                     .replace(/[ ]/g, "-")

```

```

1646     );
1647     var name = val;
1648     if (headers[val] > 0) {
1649         name = val + "-" + headers[val];
1650     }
1651     if (headers[val] == undefined) {
1652         headers[val] = 1;
1653     } else {
1654         headers[val] += 1;
1655     }
1656     node = node.wrap('<div id="' + name + '" class="anchor-wrap" ></div>');
1657     node.append(
1658         '<a class="anchor" href="#" +
1659         name +
1660         "><span class="octicon octicon-link"></span></a>'
1661     );
1662 });
1663 });
1664
1665 buttonsClickOnEnter();
1666 searchUsers();
1667 searchRepositories();
1668
1669 initCommentForm();
1670 initRepository();
1671 initWikiForm();
1672 initEditForm();
1673 initEditor();
1674 initOrganization();
1675 initAdmin();
1676 initCodeView();
1677
1678 // Repo clone url.
1679 if ($("#repo-clone-url").length > 0) {
1680     switch (localStorage.getItem("repo-clone-protocol")) {
1681         case "ssh":
1682             if ($("#repo-clone-ssh").click().length === 0) {
1683                 $("#repo-clone-https").click();
1684             }
1685             break;
1686         default:
1687             $("#repo-clone-https").click();
1688             break;
1689     }
1690 }
1691
1692 var routes = {
1693     "div.user.settings": initUserSettings,
1694     "div.repository.settings.collaboration": initRepositoryCollaboration,

```

```
1695     "div.webhook.settings": initWebhookSettings
1696 };
1697
1698 var selector;
1699 for (selector in routes) {
1700     if ($(selector).length > 0) {
1701         routes[selector]();
1702         break;
1703     }
1704 }
1705 });
1706
1707 function changeHash(hash) {
1708     if (history.pushState) {
1709         history.pushState(null, null, hash);
1710     } else {
1711         location.hash = hash;
1712     }
1713 }
1714
1715 function deSelect() {
1716     if (window.getSelection) {
1717         window.getSelection().removeAllRanges();
1718     } else {
1719         document.selection.empty();
1720     }
1721 }
1722
1723 function selectRange($list, $select, $from) {
1724     $list.removeClass("active");
1725     if ($from) {
1726         var a = parseInt($select.attr("rel").substr(1));
1727         var b = parseInt($from.attr("rel").substr(1));
1728         var c;
1729         if (a != b) {
1730             if (a > b) {
1731                 c = a;
1732                 a = b;
1733                 b = c;
1734             }
1735             var classes = [];
1736             for (var i = a; i <= b; i++) {
1737                 classes.push(".L" + i);
1738             }
1739             $list.filter(classes.join(",")).addClass("active");
1740             changeHash("#L" + a + "-" + "L" + b);
1741             return;
1742         }
1743     }
```

```

1744     $select.addClass("active");
1745     changeHash("#" + $select.attr("rel"));
1746 }
1747
1748 $(function() {
1749     if ($.user.signin).length > 0) return;
1750     $("form").areYouSure();
1751 });
1752
1753 // getByteLen counts bytes in a string's UTF-8 representation.
1754 function getByteLen(normalVal) {
1755     // Force string type
1756     normalVal = String(normalVal);
1757
1758     var byteLen = 0;
1759     for (var i = 0; i < normalVal.length; i++) {
1760         var c = normalVal.charCodeAt(i);
1761         byteLen +=
1762             c < 1 << 7
1763             ? 1
1764             : c < 1 << 11
1765             ? 2
1766             : c < 1 << 16
1767             ? 3
1768             : c < 1 << 21
1769             ? 4
1770             : c < 1 << 26
1771             ? 5
1772             : c < 1 << 31
1773             ? 6
1774             : Number.NaN;
1775     }
1776     return byteLen;
1777 }
1778
1779 function showMessageMaxLength(maxLen, textElemId, counterId) {
1780     var $msg = $("#" + textElemId);
1781     $("#" + counterId).html(maxLen - getByteLen($msg.val()));
1782
1783     var onMessageKey = function(e) {
1784         var $msg = $(this);
1785         var text = $msg.val();
1786         var len = getByteLen(text);
1787         var remainder = maxLen - len;
1788
1789         if (len >= maxLen) {
1790             $msg.val($msg.val().substr(0, maxLen));
1791             remainder = 0;
1792         }

```

```
1793
1794     $("#" + counterId).html(remainder);
1795 };
1796
1797     $msg.keyup(onMessageKey).keydown(onMessageKey);
1798 }
```

