


2


Gruntfile.js

View



358358359360361362363364


@@ -358,7 +358,7 @@ module.exports = function ( grunt ) {  
sub: true,  
boss: true,  
eqnull: true  
- },  
+ }  
// No test or globals provided - comment out to silence warnings  
// test: [  
// '=<%= app\_files.jsunit %>'



68

src/app/api/models/helpdesk-session.coffee

View



...12345678910111213141516171819202122232425262728293031323334353637383940414243444546

...  
+angular.module("doubtfire.api.models.helpdesk-session", [])  
+  
+ # API endpoint for helpdesk sessions of staff working  
+  
+.factory("HelpdeskSession", (resourcePlus) ->  
+ HelpdeskSession = resourcePlus "/helpdesk/sessions/:id", { id: "@id" }  
+  
+ #  
+ # Clocks on a user for the specified duration (in hours) given. Durations  
+ # must be specified in decimal format. E.g., 1h 15m == 1.25 hrs  
+ #  
+ HelpdeskSession.clockOn = (userId, duration, callback) ->  
+ clockOffTime = moment().add(duration, 'hours').format()  
+ dataToPost =  
+ clock\_off\_time: clockOffTime  
+ user\_id: userId  
+ onSuccess = (response) -> callback(null, response)  
+ onFailure = (response) -> callback(response)  
+ HelpdeskSession.create(dataToPost).\$promise.then(onSuccess, onFailure)  
+  
+ #  
+ # Gets the active working session of a specific userId at the helpdesk  
+ #  
+ HelpdeskSession.getActiveSessions = (userId, callback) ->  
+ onSuccess = (response) -> callback(null, response)  
+ onFailure = (response) -> callback(response)  
+ HelpdeskSession.query({ user\_id: userId, is\_active: true }).\$promise.then(onSuccess, onFailure)  
+  
+ #  
+ # Gets the list of all tutors currently working  
+ #  
+ HelpdeskSession.tutorsWorkingNow = (callback) ->  
+ onSuccess = (response) -> callback(null, response)  
+ onFailure = (response) -> callback(response)  
+ HelpdeskSession.query({ is\_active: true }).\$promise.then(onSuccess, onFailure)  
+  
+ #  
+ # Prematurely clocks off a session before it's default expiry time  
+ #  
+ HelpdeskSession.clockOffSession = (sessionId, callback) ->  
+ onSuccess = (response) -> callback(null, response)  
+ onFailure = (response) -> callback(response)  
+ HelpdeskSession.delete({ id: sessionId }).\$promise.then(onSuccess, onFailure)  
+  
+ #  
+ # Clocks off this session prematurely

```
47 + #
48 + HelpdeskSession.prototype.clockOff = (callback) ->
49 +   HelpdeskSession.clockOffSession(this.id, callback)
50 +
51 + #
52 + # Returns the duration until the session expires
53 + #
54 + HelpdeskSession.prototype.timeUntilFinish = ->
55 +   clockOffTime = moment(this.clock_off_time)
56 +   moment.duration(clockOffTime.diff())
57 +
58 + #
59 + # Returns the currently working session for the user id provided, or
60 + # null if no such session exists
61 + #
62 + HelpdeskSession.currentWorkingSession = (userId, callback) ->
63 +   HelpdeskSession.getActiveSessions userId, (error, success) ->
64 +     return callback(error) if error
65 +     callback(null, success[0])
66 +
67 + HelpdeskSession
68 +)
```

21 src/app/api/models/helpdesk-stats.coffee

View



```
...  ... @@ -0,0 +1,21 @@
1 +angular.module("doubtfire.api.models.helpdesk-stats", [])
2 +#
3 +# API endpoint for getting helpdesk stats
4 +#
5 +.factory("HelpdeskStats", ($http, $rootScope, $interval, api) ->
6 +   HelpdeskStats = {}
7 +
8 +   #
9 +   # Gets the helpdesk statistics from the time specified by `from` to the
10 +   # time specified by `to`.
11 +   #
12 +   # If to is null, statistics will fetch to now.
13 +   # If from is null, all statistics from the start of time are returned.
14 +   #
15 +   HelpdeskStats.get = (from, to, interval, callback) ->
16 +     onSuccess = (response) -> callback(null, response.data)
17 +     onFailure = (response) -> callback(response)
18 +     $http.get("#{api}/helpdesk/stats/dashgraph").then(onSuccess, onFailure)
19 +
20 +   HelpdeskStats
21 +)
```

103 src/app/api/models/helpdesk-ticket.coffee

View



```
...  ... @@ -0,0 +1,103 @@
1 +angular.module("doubtfire.api.models.helpdesk-ticket", [])
2 +#
3 +# API endpoint for serving helpdesk tickets
4 +#
5 +.factory("HelpdeskTicket", (resourcePlus) ->
6 +   HelpdeskTicket = resourcePlus "/helpdesk/tickets/:id", { id: "@id" }
7 +
8 +   #
9 +   # Submits a new helpdesk ticket. Provide a project ID, description and task
10 +   # def id. Project ID is required, whereas the others can be `null`. Callback
11 +   # is a function whose first parameter is error and second parameter is success
12 +   # data.
```

```
13 + #
14 + HelpdeskTicket.submitTicket = (projectId, description, taskDefinitionId, callback) ->
15 +   dataToPost =
16 +     project_id: projectId
17 +     description: description
18 +     task_definition_id: taskDefinitionId
19 +   onSuccess = (response) -> callback(null, response)
20 +   onFailure = (response) -> callback(response)
21 +   HelpdeskTicket.create(dataToPost)
22 +     .$.promise
23 +     .then(onSuccess, onFailure)
24 +
25 + #
26 + # Gets all tickets by a specific state, optionally for the specified user id.
27 + # If user is not specified, then all tickets will be returned.
28 + #
29 + HelpdeskTicket.getTicketsByState = (state, userId, callback) ->
30 +   onSuccess = (response) -> callback(null, response)
31 +   onFailure = (response) -> callback(response)
32 +   HelpdeskTicket.query({ user_id: userId, shallow: userId?, filter: state })
33 +     .$.promise
34 +     .then(onSuccess, onFailure)
35 +
36 + #
37 + # Gets all unresolved tickets (optionally by user)
38 + #
39 + HelpdeskTicket.getUnresolvedTickets = (userId, callback) ->
40 +   HelpdeskTicket.getTicketsByState('unresolved', userId, callback)
41 +
42 + #
43 + # Gets all resolved tickets (optionally by user)
44 + #
45 + HelpdeskTicket.getResolvedTickets = (userId, callback) ->
46 +   HelpdeskTicket.getTicketsByState('resolved', userId, callback)
47 +
48 + #
49 + # Gets all closed tickets (optionally by user)
50 + #
51 + HelpdeskTicket.getClosedTickets = (userId, callback) ->
52 +   HelpdeskTicket.getTicketsByState('closed', userId, callback)
53 +
54 + #
55 + # Closes a specified ticket with the given ID
56 + #
57 + HelpdeskTicket.closeTicket = (ticketId, callback) ->
58 +   onSuccess = (response) -> callback(null, response)
59 +   onFailure = (response) -> callback(response)
60 +   HelpdeskTicket.delete({ id: ticketId })
61 +     .$.promise
62 +     .then(onSuccess, onFailure)
63 +
64 + #
65 + # Prototype close ticket method
66 + #
67 + HelpdeskTicket.prototype.close = (callback) ->
68 +   HelpdeskTicket.closeTicket(this.id, callback)
69 +
70 + #
71 + # Resolves a specified ticket with the given ID
72 + #
73 + HelpdeskTicket.resolveTicket = (ticketId, callback) ->
74 +   onSuccess = (response) -> callback(null, response)
75 +   onFailure = (response) -> callback(response)
```

```
76 +   HelpdeskTicket.delete({ id: ticketId, resolve: true })
77 +   .$.promise
78 +   .then(onSuccess, onFailure)
79 +
80 + #
81 + # Prototype resolve ticket method
82 + #
83 + HelpdeskTicket.prototype.resolve = (callback) ->
84 +   HelpdeskTicket.resolveTicket(this.id, callback)
85 +
86 + #
87 + # Returns the duration since the ticket was open
88 + #
89 + HelpdeskTicket.prototype.lengthOfTimeOpen = ->
90 +   createdTime = moment(this.created_at)
91 +   moment.duration(createdTime.diff())
92 +
93 + #
94 + # Checks if the user with the specified ticket has a ticket open already
95 + # and if so returns that ticket in the callback
96 + #
97 + HelpdeskTicket.currentOpenTicket = (userId, callback) ->
98 +   HelpdeskTicket.getUnresolvedTickets(userId, (error, success) ->
99 +     return callback(error) if error
100 +     callback(null, success[0])
101 +
102 + HelpdeskTicket
103 +)
```

3 src/app/api/models/models.coffee

View



```
@@ -8,6 +8,9 @@ angular.module("doubtfire.api.models", [
8   "doubtfire.api.models.tutorial"
9   "doubtfire.api.models.learning-alignments"
10  "doubtfire.api.models.intended-learning-outcome"
11  + "doubtfire.api.models.helpdesk-ticket"
12  + "doubtfire.api.models.helpdesk-session"
13  + "doubtfire.api.models.helpdesk-stats"
14  "doubtfire.api.models.task"
15  "doubtfire.api.models.task-comment"
16  "doubtfire.api.models.task-definition"
```

1 src/app/app.coffee

View



```
@@ -16,6 +16,7 @@ angular.module('doubtfire', [
16  'doubtfire.tasks'
17  'doubtfire.projects'
18  'doubtfire.users'
19  + 'doubtfire.helpdesk'
20  'doubtfire.groups'
21  'doubtfire.visualisations'
22  ])
```

4 src/app/common/header/header.coffee

View



```
@@ -1,7 +1,9 @@
1  #
2  # Controllers and providers related to the header/nav bar
3  #
4  -angular.module('doubtfire.common.header', [])
4  +angular.module('doubtfire.common.header', [
5  +  'doubtfire.common.header.helpdesk-header'
```

```
6 +))
5 7
6 8 .controller("BasicHeaderCtrl", ($scope, $state, $modal, User, AboutDoubtfireModal, UserNotificationSettingsModal, UserS
7 9 $scope.menus = headerService.getMenus())
```

30 src/app/common/header/header.scss

View



```
@@ -5,6 +5,16 @@
5 5 border-right: none;
6 6 margin-bottom:0;
7 7
8 8 + // Mobile responsive fixes
9 9 + @media (min-width: $screen-md) {
10 10 + // Fix for dynamic content
11 11 + .navbar-right {
12 12 + width: 80%;
13 13 + display: flex;
14 14 + justify-content: flex-end;
15 15 + }
16 16 + }
17 17 +
18 18 .dropdown-units {
19 19 li:not(.divider) a {
20 20 padding-top: 0.5em;
@@ -18,6 +28,8 @@
18 28 padding-left: 1.75ex;
19 29 flex: 1;
20 30 text-align: right;
21 31 + overflow: hidden;
22 32 + text-overflow: ellipsis;
23 33 }
24 34 }
25 35 }
@@ -35,3 +47,21 @@ a.navbar-brand {
35 47 margin: 0;
36 48 }
37 49 }
50 50 +
51 51 + @media (max-width: $screen-md) {
52 52 + .navbar {
53 53 + position: fixed;
54 54 + top: 0;
55 55 + left: 0;
56 56 + right: 0;
57 57 + z-index: 9;
58 58 + }
59 59 + .navbar .navbar-nav .open ul.dropdown-menu {
60 60 + background-color: #fcfcfc;
61 61 + border-bottom: 1px solid #aaa;
62 62 + border-radius: 0;
63 63 + }
64 64 + body {
65 65 + padding-top: $navbar-height;
66 66 + }
67 67 +}
```

13 src/app/common/header/header.tpl.html

View



```
@@ -15,15 +15,10 @@ <h1 class="logo">Doubtfire</h1>
15 15 <!-- Collect the nav links, forms, and other content for toggling -->
16 16 <div class="collapse navbar-collapse" ng-class="!navCollapsed && 'in'" ng-show="!navCollapsed">
```

```

17 17      <ul class="nav navbar-nav navbar-right">
18      -      <li class="dropdown dropdown-admin" dropdown ng-repeat="menu in menus | orderBy:'name'">
19      -      <a class="dropdown-toggle" dropdown-toggle role="button"><span class="glyphicon glyphicon-{{menu.icon}}">{{menu
20      -      <ul class="dropdown-menu">
21      -      <li ng-repeat="link in menu.links | orderBy:'name'"
22      -      ng-class="{active: link.class === 'active'}">
23      -      <a href="{{link.url}}">{{link.name}}</a>
24      -      </li>
25      -      </ul>
26      -      </li>
18 +      <helpdesk-header
19 +      projects="projects"
20 +      unit-roles="unitRoles">
21 +      </helpdesk-header>
27 22
28 23      <li class="dropdown dropdown-units" dropdown ng-hide="projects.length == 0 && unitRoles.length == 0">
29 24      <a role="button" class="dropdown-toggle" dropdown-toggle><span class="glyphicon glyphicon-book"></span> Units <

```

57 src/app/common/header/helpdesk-header/helpdesk-header.coffee

View



```

...  ...  @@ -0,0 +1,57 @@
1  +#
2  +# Directive for controlling helpdesk dropdown
3  +#
4  +angular.module('doubtfire.common.header.helpdesk-header', [])
5  +.directive 'helpdeskHeader', ->
6  +  restrict: 'E'
7  +  replace: true
8  +  scope:
9  +    projects: '='
10 +    unitRoles: '='
11 +  templateUrl: 'common/header/helpdesk-header/helpdesk-header.tpl.html'
12 +  controller: ($scope, $rootScope, currentUser, HelpdeskTicketModal, HelpdeskSessionModal, HelpdeskTicket, HelpdeskSess
13 +    #
14 +    # This function updates the visibility of the element
15 +    #
16 +    updateVisibility = ->
17 +    # These variables help us know if the user teaches or studies a unit
18 +    $scope.studiesUnit = $scope.projects?.length > 0
19 +    $scope.teachesUnit = $scope.unitRoles?.length > 0
20 +    # Only show the helpdesk dropdown if I'm learning or teaching a unit
21 +    $scope.showHelpdeskDropdown = $scope.studiesUnit or $scope.teachesUnit
22 +    # Check if the user has a ticket|session open to switch which modals they can open
23 +    if $scope.studiesUnit
24 +      HelpdeskTicket.currentOpenTicket $scope.currentUser.id, (error, data) ->
25 +        $scope.currentOpenTicket = data
26 +    if $scope.teachesUnit
27 +      HelpdeskSession.currentWorkingSession $scope.currentUser.id, (error, data) ->
28 +        $scope.currentWorkingSession = data
29 +
30 +
31 +    #
32 +    # Watch for changes in projects and unitRoles length's and update the
33 +    # visibility in case they change
34 +    #
35 +    $scope.$watch('projects.length', updateVisibility)
36 +    $scope.$watch('unitRoles.length', updateVisibility)
37 +
38 +    # Current user binding
39 +    $scope.currentUser = currentUser.profile
40 +
41 +    # Watch the root scope for changes to the current ticket|session and set if needed

```

```
42 +   $rootScope.$on 'CurrentOpenTicket', (event, ticket) ->
43 +     $scope.currentOpenTicket = ticket
44 +   $rootScope.$on 'CurrentWorkingSession', (event, session) ->
45 +     $scope.currentWorkingSession = session
46 +
47 +   #
48 +   # Opens the submit ticket modal
49 +   #
50 +   $scope.openHelpdeskTicketModal = ->
51 +     HelpdeskTicketModal.show $scope.currentOpenTicket
52 +
53 +   #
54 +   # Opens the session modal
55 +   #
56 +   $scope.openHelpdeskSessionModal = ->
57 +     HelpdeskSessionModal.show $scope.currentWorkingSession
```

20 src/app/common/header/helpdesk-header/helpdesk-header.scss

View



```
...   ... @@ -0,0 +1,20 @@
1 +// Mobile responsive fixes
2 +@media (min-width: $screen-md) {
3 +  .helpdesk.header {
4 +    display: inline;
5 +    .helpdesk-direct-link {
6 +      padding-right: 0;
7 +      .badge {
8 +        background-color: $doubtfire-color;
9 +      }
10 +    }
11 +    .dropdown-toggle {
12 +      float: right;
13 +    }
14 +  }
15 +}
16 +@media (max-width: $screen-md) {
17 +  .helpdesk.header .helpdesk-direct-link {
18 +    display: none;
19 +  }
20 +}
```

33 src/app/common/header/helpdesk-header/helpdesk-header.tpl.html

View



```
...   ... @@ -0,0 +1,33 @@
1 +<li dropdown class="helpdesk header" ng-show="showHelpdeskDropdown">
2 +  <a class="helpdesk-direct-link pull-left" ng-click="openHelpdeskTicketModal()" ng-if="currentOpenTicket">
3 +    <span class="badge" tooltip="Ticket open for {{currentOpenTicket.lengthOfTimeOpen().humanize()}}" tooltip-placement="top">
4 +      <i class="fa fa-ticket"></i>
5 +    </span>
6 +  </a>
7 +  <a class="helpdesk-direct-link pull-left" ng-click="openHelpdeskSessionModal()" ng-if="currentWorkingSession">
8 +    <span class="badge" tooltip="You finish working in {{currentWorkingSession.timeUntilFinish().humanize()}}" tooltip-placement="top">
9 +      <i class="fa fa-clock-o"></i>
10 +    </span>
11 +  </a>
12 +  <a class="dropdown-toggle" dropdown-toggle>
13 +    <i class="fa fa-ambulance"></i>
14 +    Helpdesk
15 +    <b class="caret"></b>
16 +  </a>
17 +  <ul class="dropdown-menu">
18 +    <li ng-if="studiesUnit">
19 +      <a ng-click="openHelpdeskTicketModal()">
```

```

20 +         {{ currentOpenTicket ? "View Open Ticket" : "Submit a Ticket" }}
21 +       </a>
22 +     </li>
23 +     <li ng-if="teachesUnit">
24 +       <a ng-click="openHelpdeskSessionModal()">
25 +         {{ currentWorkingSession ? "Clock Off" : "Clock On" }}
26 +       </a>
27 +     </li>
28 +     <li class="divider"></li>
29 +     <li>
30 +       <a ui-sref="helpdesk"> View Helpdesk Dashboard </a>
31 +     </li>
32 + </ul>
33 +</li>

```

20 src/app/common/modals/confirmation-modal/confirmation-modal.tpl.html

View 

```

@@ -6,17 +6,17 @@ <h3 class="modal-title">
6      <small>Please confirm that you want to perform this action.</small>
7    </h3>
8  </div>
9  - <div class="modal-body">
10 -   {{message}}
11 + <div class="modal-body clearfix">
12 +   <p class="col-sm-12">
13 +     {{message}}
14 +   </p>
15 + </div>
16   <div class="modal-footer clearfix">
17   - <div class="btn-group">
18   -   <label class="btn btn-success" ng-click="confirmAction()">
19   -     <i class="fa fa-check"></i> Confirm
20   -   </label>
21   -   <label class="btn btn-danger" ng-click="cancelAction()">
22   -     <i class="fa fa-times"></i> Cancel
23   -   </label>
24   - </div>
25 +   <button class="btn btn-danger" ng-click="cancelAction()">
26 +     <i class="fa fa-times"></i> Cancel
27 +   </button>
28 +   <button class="btn btn-success" ng-click="confirmAction()">
29 +     <i class="fa fa-check"></i> Confirm
30 +   </button>
31 + </div>
32 </div>

```

10 src/app/common/services/project-service.coffee

View 

```

@@ -26,7 +26,7 @@ angular.module("doubtfire.common.services.projects", [])
26     projectService.loadedProjects = null
27
28     projectService.getProjects = ( callback ) ->
29     -   if ! projectService.loadedProjects?
30     +   unless projectService.loadedProjects?
31       projectService.loadedProjects = []
32       Project.query(
33         (projects) ->
34
35 @@ -43,6 +43,14 @@ angular.module("doubtfire.common.services.projects", [])
43     projectService.loadedProjects
44
45
46 + #
47 + # Finds a project with the given id

```



```

48 + #
49 + projectService.findProject = (id, callback) ->
50 +   projectService.getProjects (projects) ->
51 +     projFound = _.find(projects, (p) -> p.project_id is id)
52 +     callback(projFound)
53 +
46 54   ###
47 55   projects's can update their task stats
48 56   converts the | delimited stats to its component arrays

```

2 src/app/common/services/unit-service.coffee

View



```

193 193   (response) ->
194 194     alertService.add("danger", "Grade was not updated: #{response.data.error}", 8000)
195 195
196 196   - projectService.updateTaskStats(student, student.stats)
197 197   + projectService.updateTaskStats(student, student.stats) if student.stats
198 198   projectService.addTaskDetailsToProject(student, unit)
199 199   unit.getGroups = (group_set, group_callback) ->

```

33 src/app/helpdesk/helpdesk-ticket/helpdesk-ticket.coffee

View



```

... 1 +angular.module('doubtfire.helpdesk.helpdesk-ticket', [])
2 +#
3 +# Directive to display information related to a helpdesk ticket
4 +#
5 +.directive('helpdeskTicket', ->
6 +  restrict: 'E'
7 +  templateUrl: 'helpdesk/helpdesk-ticket/helpdesk-ticket.tpl.html'
8 +  replace: true
9 +  scope:
10 +    ticket: '='
11 +  controller: ($scope, HelpdeskTicket, projectService, unitService, HelpdeskTicketModal) ->
12 +
13 +    #
14 +    # Returns warning style for ticket
15 +    #
16 +    $scope.warningStyle = ->
17 +      minsWaiting = -$scope.ticket.lengthOfTimeOpen().asMinutes()
18 +      if minsWaiting < 3
19 +        'success'
20 +      else if 3 <= minsWaiting < 6
21 +        'primary'
22 +      else if 6 <= minsWaiting < 9
23 +        'warning'
24 +      else if minsWaiting >= 9
25 +        'danger'
26 +
27 +    #
28 +    # Opens the submit ticket modal
29 +    #
30 +    $scope.openHelpdeskTicketModal = ->
31 +      return if $scope.isLoading
32 +      HelpdeskTicketModal.show $scope.ticket
33 +)

```

49 src/app/helpdesk/helpdesk-ticket/helpdesk-ticket.scss

View



```
... @@ -0,0 +1,49 @@
1 +.helpdesk-ticket {
2 + $br: 8px;
3 + font-size: 1.3em;
4 + // Mobile responsive fixes
5 + @media (max-width: $screen-md) {
6 +   margin-bottom: 1em;
7 +   .unit, .unit-name, .task-name {
8 +     text-align: center !important;
9 +     width: 100% !important;
10 +   }
11 + }
12 + header, .content {
13 +   padding: 8px;
14 + }
15 + header {
16 +   background-color: $doubtfire-color;
17 +   color: #fff;
18 +   font-size: 1.25em;
19 +   border-radius: $br $br 0 0;
20 +   line-height: 1.25;
21 + }
22 + .content {
23 +   color: #525252;
24 +   background-color: #ecedef;
25 +   .unit-name {
26 +     text-overflow: ellipsis;
27 +     width: 65%;
28 +     text-align: right;
29 +     overflow: hidden;
30 +     white-space: nowrap;
31 +   }
32 + }
33 + .content .loading {
34 +   text-align: center;
35 +   font-size: 2em;
36 + }
37 + hr {
38 +   border: 1px dashed grey;
39 + }
40 + border: 2px solid #ecedef;
41 + border-radius: $br;
42 + transition: all 0.25s ease;
43 + &:hover {
44 +   position: relative;
45 +   top: -10px;
46 +   cursor: pointer;
47 +   box-shadow: 0px 0px 10px 0px rgba(0,0,0,0.4);
48 + }
49 +}
```

29 src/app/helpdesk/helpdesk-ticket/helpdesk-ticket.tpl.html

View



```
... @@ -0,0 +1,29 @@
1 +<div class="helpdesk-ticket" ng-click="openHelpdeskTicketModal()">
2 +   <header ng-if="isLoading">
3 +     <strong>Doubtfire</strong> Helpdesk Ticket
4 +   </header><!--/ticket-header-->
5 +   <header ng-if="!isLoading">
6 +     <strong>{{ticket.project.student_name}}</strong> <span class="pull-right">#{{ticket.position}}</span>
7 +   </header><!--/ticket-header-->
8 +   <div class="content">
9 +     <div class="loading" ng-if="isLoading">
```

```

10 +     <i class="fa fa-spinner fa-pulse"></i>
11 + </div><!--/loading-ticket-->
12 + <div ng-if="!isLoading">
13 +     <div class="unit">
14 +         <label class="label label-info">{{ticket.project.unit_code}}</label>
15 +         <strong class="unit-name pull-right">{{ticket.project.unit_name}}</strong>
16 +     </div>
17 +     <div class="task-name text-right">
18 +         {{ticket.task_definition.name}}
19 +     </div>
20 +     <hr>
21 +     <div class="text-center">
22 +         Waiting for
23 +         <strong class="text-{{warningStyle()}}">
24 +             {{ticket.lengthOfTimeOpen().humanize()}}
25 +         </strong>
26 +     </div>
27 + </div><!--/data-loaded-->
28 + </div><!--/ticket-content-->
29 +</div>

```

5 src/app/helpdesk/helpdesk.coffee

View



```

... @@ -0,0 +1,5 @@
1 +angular.module('doubtfire.helpdesk', [
2 +   'doubtfire.helpdesk.modals'
3 +   'doubtfire.helpdesk.states'
4 +   'doubtfire.helpdesk.helpdesk-ticket'
5 +])

```

4 src/app/helpdesk/modals/modals.coffee

View



```

... @@ -0,0 +1,4 @@
1 +angular.module('doubtfire.helpdesk.modals', [
2 +   'doubtfire.helpdesk.modals.ticket-modal'
3 +   'doubtfire.helpdesk.modals.session-modal'
4 +])

```

81 src/app/helpdesk/modals/session-modal/session-modal.coffee

View



```

... @@ -0,0 +1,81 @@
1 +#
2 +# The helpdesk session modal is the modal by which staff can clock on and clock
3 +# off from the helpdesk
4 +#
5 +angular.module('doubtfire.helpdesk.modals.session-modal', [])
6 +
7 +.factory('HelpdeskSessionModal', ($modal) ->
8 +   HelpdeskSessionModal = {}
9 +
10 +   #
11 +   # Pass in the user, project
12 +   #
13 +   HelpdeskSessionModal.show = (session) ->
14 +     $modal.open
15 +     templateUrl: 'helpdesk/modals/session-modal/session-modal.tpl.html'
16 +     controller: 'HelpdeskSessionModal'
17 +     resolve:
18 +       session: -> session
19 +
20 +   HelpdeskSessionModal
21 +)
22 +

```

```
23 +.controller('HelpdeskSessionModal', ($scope, $state, $rootScope, $interval, $modalInstance, HelpdeskSession, Confirmati
24 + $scope.clockOffState = session? # I can clock off if I was given a session
25 + $scope.clockOnState = !session? # I can clock on if I wasn't given a session
26 +
27 + # Strip the profile
28 + $scope.currentUser = currentUser.profile
29 +
30 + # Duration to clock on
31 + $scope.duration = { hours: null }
32 +
33 + #
34 + # Updates the automatic clock off time
35 + #
36 + updateAutomaticClockOffTime = (newDuration) ->
37 +   return unless newDuration?
38 +   $scope.automaticClockOffTime =
39 +     moment().add($scope.duration.hours, 'hours')
40 +
41 + # Watch duration to calculate the automaticClockOffTime
42 + $scope.$watch 'duration.hours', updateAutomaticClockOffTime
43 +
44 + # If we have a session, work out the automatic clock off time
45 + if $scope.clockOffState
46 +   $scope.automaticClockOffTime = moment(session.clock_off_time)
47 +   updateTimeLeft = ->
48 +     $scope.timeLeft = moment.duration($scope.automaticClockOffTime.diff())
49 +   updateTimeLeft()
50 +   updateTimeLeftInterval = $interval updateTimeLeft, 1000
51 +   # Must explicitly destroy the time interval when the controller ends!
52 +   $scope.$on '$destroy', ->
53 +     $interval.cancel(updateTimeLeftInterval)
54 +
55 + #
56 + # Clocks on the session
57 + #
58 + $scope.clockOn = ->
59 +   clockOnCallback = (error, success) ->
60 +     if success
61 +       $modalInstance.close(success)
62 +       $rootScope.$broadcast('CurrentWorkingSession', success)
63 +       alertService.add("success", "Clocked on successfully.", 2000)
64 +     if error
65 +       alertService.add("danger", "Error: #{error.data.error}", 6000)
66 +   HelpdeskSession.clockOn(currentUser.id, $scope.duration.hours, clockOnCallback)
67 +
68 + #
69 + # Clocks off the session
70 + #
71 + $scope.clockOff = ->
72 +   clockOffCallback = (error, success) ->
73 +     if success
74 +       $modalInstance.close(success)
75 +       $rootScope.$broadcast('CurrentWorkingSession', null)
76 +       alertService.add("success", "You have been clocked off.", 2000)
77 +     if error
78 +       alertService.add("danger", "Error: #{error.data.error}", 6000)
79 +   ConfirmationModal.show 'Clock off now', "You still have #{session.timeUntilFinish().humanize()} left to work. Are y
80 +   session.clockOff(clockOffCallback)
81 +)
```

8 src/app/helpdesk/modals/session-modal/session-modal.scss

View



... @@ -0,0 +1,8 @@

```
1 +.session-modal {
2 +  .duration-entry {
3 +    margin: 1em auto;
4 +  }
5 +  .time-left {
6 +    font-size: 3em;
7 +  }
8 +}
```

62 src/app/helpdesk/modals/session-modal/session-modal.tpl.html

View



```
... @@ -0,0 +1,62 @@
1 +<div class="session-modal">
2 +  <form class="form-horizontal" name="sessionForm" role="form">
3 +    <div class="modal-header">
4 +      <h3>Clock {{clockOnState ? 'on to' : 'off from'}} the helpdesk</h3>
5 +    </div><!--/new-ticket-header-->
6 +    <div class="modal-body text-center clearfix">
7 +      <div class="col-sm-12" ng-if="clockOnState">
8 +        <label>
9 +          Please enter how long you are working at the helpdesk for.
10 +        </label>
11 +        <div class="input-group duration-entry col-sm-4">
12 +          <input name="duration"
13 +            ng-model="duration.hours"
14 +            class="form-control input-lg"
15 +            type="number"
16 +            placeholder="2"
17 +            step=".25"
18 +            min="0.25"
19 +            max="8"
20 +            required>
21 +          <span class="input-group-addon">hours</span>
22 +        </div>
23 +        <p class="text-danger" ng-show="sessionForm.duration.$error.number">
24 +          Please enter duration as a decimal number. For example, 1 hour and
25 +          15 minutes would be represented as <code>1.25</code> hours.
26 +        </p>
27 +        <p class="text-danger" ng-show="sessionForm.duration.$error.min">
28 +          You must work for at least fifteen minutes.
29 +        </p>
30 +        <p class="text-danger" ng-show="sessionForm.duration.$error.max">
31 +          You can't work more than eight hours.
32 +        </p>
33 +        <p class="help-block" ng-show="sessionForm.duration.$valid">
34 +          You will be automatically clocked off at <strong>{{automaticClockOffTime.format('hh:mm a')}}</strong>,
35 +          but you can manually clock off at an earlier time.
36 +        </p>
37 +      </div><!--/clock-on-state-->
38 +      <div class="col-sm-12" ng-if="clockOffState">
39 +        <label>
40 +          You finish working at the helpdesk in
41 +        </label>
42 +        <div class="time-left">
43 +          {{timeLeft.hours()}}h {{timeLeft.minutes()}}m {{timeLeft.seconds()}}s
44 +        </div>
45 +        <p class="help-block">
46 +          (That's at {{automaticClockOffTime.format('hh:mm a')}})
47 +        </p>
48 +      </div>
49 +    </div>
50 +    <div class="modal-footer text-right">
51 +      <button ng-click="$dismiss()" class="btn btn-primary">
```

```

52 +         Close
53 +     </button>
54 +     <button ng-disabled="!sessionForm.$valid" ng-click="clockOn()" class="btn btn-success" ng-if="clockOnState">
55 +         <i class="fa fa-clock-o"></i> <i class="fa fa-sign-in"></i> Clock On
56 +     </button>
57 +     <button ng-click="clockOff()" class="btn btn-danger" ng-if="clockOffState">
58 +         <i class="fa fa-clock-o"></i> <i class="fa fa-sign-out"></i> Clock Off Now
59 +     </button>
60 + </div>
61 + </form>
62 +</div>

```

107 src/app/helpdesk/modals/ticket-modal/ticket-modal.coffee

View



```

... @@ -0,0 +1,107 @@
1  +#
2  +# The helpdesk ticket modal is the modal by which students can submit
3  +# new tickets to or review their ticket information
4  +#
5  +angular.module('doubtfire.helpdesk.modals.ticket-modal', [])
6  +
7  +.factory('HelpdeskTicketModal', ($modal) ->
8  +  HelpdeskTicketModal = {}
9  +
10 +  #
11 +  # Pass in the user, project
12 +  #
13 +  HelpdeskTicketModal.show = (ticket) ->
14 +    $modal.open
15 +      templateUrl: 'helpdesk/modals/ticket-modal/ticket-modal.tpl.html'
16 +      controller: 'HelpdeskTicketModal'
17 +      resolve:
18 +        ticket: -> ticket
19 +
20 +  HelpdeskTicketModal
21 +)
22 +
23 +.controller('HelpdeskTicketModal', ($scope, $state, $rootScope, $modalInstance, HelpdeskTicket, ConfirmationModal, alert) ->
24 +  $scope.isNew = !ticket?
25 +  $scope.ticket = ticket || new HelpdeskTicket()
26 +
27 +  #
28 +  # Callback when a task definition has changed
29 +  #
30 +  $scope.taskDefSelected = (taskDef) ->
31 +    $scope.ticket.task_definition = taskDef
32 +
33 +  # Use project service to get projects
34 +  unless $scope.ticket.project?
35 +    projectService.getProjects (projects) ->
36 +      $scope.projects = projects
37 +      $scope.ticket.project = projects[0] if projects.length == 1
38 +      unless $scope.isNew
39 +        # Use the ticket information to find the correct project
40 +        projectService.findProject $scope.ticket.project_id, (p) ->
41 +          $scope.ticket.project = p
42 +
43 +  #
44 +  # Watch when a project is changed to update the selected unit
45 +  #
46 +  $scope.$watch 'ticket.project.project_id', (newId) ->
47 +    return unless newId?
48 +    $scope.taskDefSelected null if $scope.isNew # reset which task selected

```

```

49 +     unitService.getUnit $scope.ticket.project.unit_id, false, false, (response) ->
50 +         $scope.ticket.unit = response
51 +     # If ticket was provided, we need to look up the task def now from the
52 +     # unit loaded if it has one
53 +     if not $scope.isNew and $scope.ticket.task_definition_id?
54 +         taskDef = $scope.ticket.unit.taskDef($scope.ticket.task_definition_id)
55 +         $scope.taskDefSelected taskDef
56 +
57 + #
58 + # Sends a HTTP request to open a ticket
59 + #
60 + $scope.openTicket = ->
61 +     openTicketCallback = (error, success) ->
62 +         if success
63 +             $modalInstance.close(success)
64 +             $rootScope.$broadcast('CurrentOpenTicket', success)
65 +             alertService.add("success", "Ticket created successfully.", 2000)
66 +         if error
67 +             alertService.add("danger", "Error: #{error.data.error}", 6000)
68 +     HelpdeskTicket.submitTicket(
69 +         $scope.ticket.project.project_id,
70 +         $scope.ticket.description,
71 +         $scope.ticket.task_definition?.id,
72 +         openTicketCallback
73 +     )
74 +
75 + #
76 + # Sends a request to resolve the ticket
77 + #
78 + $scope.resolveTicket = ->
79 +     # TODO: Resolved by whom?
80 +     resolveTicketCallback = (error, success) ->
81 +         if success
82 +             $modalInstance.close(success)
83 +             $rootScope.$broadcast('CurrentOpenTicket', null)
84 +             alertService.add("success", "Ticket resolved!", 2000)
85 +         if error
86 +             alertService.add("danger", "Error: #{error.data.error}", 6000)
87 +     $scope.ticket.resolve(resolveTicketCallback)
88 +
89 + #
90 + # Sends a request to close the ticket
91 + #
92 + $scope.closeTicket = ->
93 +     closeTicketCallback = (error, success) ->
94 +         if success
95 +             $modalInstance.close(success)
96 +             $rootScope.$broadcast('CurrentOpenTicket', null)
97 +             alertService.add("success", "Ticket closed.", 2000)
98 +         if error
99 +             alertService.add("danger", "Error: #{error.data.error}", 6000)
100 +     ConfirmationModal.show 'Close Ticket', 'Are you sure you want to close the ticket without resolving the issue?', ->
101 +         $scope.ticket.close(closeTicketCallback)
102 +
103 + #
104 + # Goes to the selected task
105 + #
106 + $scope.goToSelectedTask = $modalInstance.dismiss
107 + )

```

14 src/app/helpdesk/modals/ticket-modal/ticket-modal.scss

View



... @@ -0,0 +1,14 @@

```
1 +.ticket-modal {
2 +  .unit-selection .dropdown {
3 +    width: 100%;
4 +    button {
5 +      text-align: right;
6 +    }
7 +  }
8 +  label {
9 +    margin-right: 1ex;
10 + }
11 + .unit-selection .dropdown-menu > li > a {
12 +   padding: 1em;
13 + }
14 +}
```

103 src/app/helpdesk/modals/ticket-modal/ticket-modal.tpl.html

View 

```
... @@ -0,0 +1,103 @@
1 +<div class="ticket-modal">
2 +  <form class="form-horizontal" role="form">
3 +    <div class="modal-header" ng-show="isNew">
4 +      <h3>Submit a Ticket</h3>
5 +      <p class="lead">
6 +        Having trouble with something? Get help from a tutor at the programming
7 +        helpdesk, located at ATC620.
8 +      </p>
9 +    </div><!--/new-ticket-header-->
10 +    <div class="modal-header" ng-hide="isNew">
11 +      <h3>Show Ticket</h3>
12 +    </div><!--/existing-ticket-header-->
13 +    <div class="modal-body">
14 +      <div class="form-group unit-selection">
15 +        <label class="col-sm-3 control-label">Unit</label>
16 +        <div class="col-sm-7">
17 +          <p class="form-control-static" ng-if="projects.length == 1 || !isNew">
18 +            <label class="label label-info small">{{ticket.project.unit_code}}</label>
19 +            <span class="unit-name">{{ticket.project.unit_name}}</span>
20 +          </p>
21 +          <div class="btn-group dropdown" ng-if="projects.length > 1 && isNew" dropdown is-open="status.isopen">
22 +            <button type="button" class="btn btn-default form-control dropdown-toggle" dropdown-toggle>
23 +              <span ng-show="ticket.project" class="pull-left">
24 +                <label class="label label-info small">{{ticket.project.unit_code}}</label>
25 +                <span class="unit-name">{{ticket.project.unit_name}}</span>
26 +              </span>
27 +              <span ng-hide="ticket.project" class="pull-left">
28 +                Select a Unit
29 +              </span>
30 +              <span class="caret"></span>
31 +            </button>
32 +            <ul class="dropdown-menu">
33 +              <li ng-repeat="project in projects">
34 +                <a ng-click="ticket.project = project">
35 +                  <label class="label label-info small">{{project.unit_code}}</label>
36 +                  <span class="unit-name">{{project.unit_name}}</span>
37 +                </a>
38 +              </li>
39 +            </ul>
40 +          </div>
41 +        </div>
42 +      </div><!--/unit-selection-->
43 +      <div class="form-group" ng-if="ticket.unit">
44 +        <label class="col-sm-3 control-label">Task</label>
45 +        <div class="col-sm-7" ng-if="isNew">
```



```
46 +         <task-definition-selector button-style="default" unit="ticket.unit" ng-model="ticket.task_definition" on-select="selectTaskDefinition($event)" class="form-control-static">
47 +         <p class="help-block">
48 +             Select a task that you want help with, or leave blank if you have a
49 +             general question.
50 +         </p>
51 +     </div><!--/task-selector-if-new-->
52 +     <div class="col-sm-7" ng-if="!isNew">
53 +         <div ng-if="ticket.task_definition" class="selected-task-detail">
54 +             <label class="label label-info">{{ticket.task_definition.abbreviation}}</label>
55 +             {{ticket.task_definition.name}}
56 +         </div>
57 +         <div class="text-muted form-control-static" ng-hide="ticket.task_definition">
58 +             No task provided
59 +         </div>
60 +     </div>
61 + </div><!--/task-selection-->
62 + <div class="form-group">
63 +     <label class="col-sm-3 control-label">Description</label>
64 +     <div class="col-sm-7" ng-if="isNew">
65 +         <markdown-editor ng-model="ticket.description" placeholder="Enter description" height="100"></markdown-editor>
66 +         <p class="help-block">
67 +             Give the tutor some context to your issue by providing a description
68 +             of your question. {{description}}
69 +         </p>
70 +     </div>
71 +     <div class="col-sm-7" ng-if="!isNew">
72 +         <div ng-show="ticket.description" class="form-control-static" ng-bind-html="ticket.description | markdown"></div>
73 +         <p ng-hide="ticket.description" class="text-muted form-control-static">
74 +             No description provided
75 +         </p>
76 +     </div>
77 + </div><!--/description-->
78 + <div class="form-group" ng-if="!isNew">
79 +     <label class="col-sm-3 control-label">In Queue For</label>
80 +     <div class="col-sm-7 form-control-static">
81 +         {{ticket.lengthOfTimeOpen().humanize()}}
82 +         <small class="text-muted">
83 +             ({{ticket.created_at | date:'shortTime'}})
84 +         </small>
85 +     </div>
86 + </div>
87 + </div>
88 + <div class="modal-footer text-right">
89 +     <button ng-click="$dismiss()" class="btn btn-primary">
90 +         Close
91 +     </button>
92 +     <button ng-disabled="ticket.project == null" ng-click="openTicket()" class="btn btn-success" ng-if="isNew">
93 +         <i class="fa fa-ticket"></i> <i class="fa fa-plus"></i> Submit Ticket
94 +     </button>
95 +     <button ng-click="closeTicket()" class="btn btn-danger" ng-if="!isNew">
96 +         <i class="fa fa-ticket"></i> <i class="fa fa-trash-o"></i> Close Ticket
97 +     </button>
98 +     <button ng-click="resolveTicket()" class="btn btn-success" ng-if="!isNew" if-role="Tutor Convenor Admin">
99 +         <i class="fa fa-ticket"></i> <i class="fa fa-check"></i> Resolve Ticket
100 +     </button>
101 + </div><!--/footer-->
102 + </form>
103 + </div>
```

166 src/app/helpdesk/states/dashboard/dashboard.coffee

View



```
... @@ -0,0 +1,166 @@
1  +#
```

```

2  +# The programming helpdesk dashboard
3  +#
4  +angular.module('doubtfire.helpdesk.states.dashboard', [])
5  +.config((headerServiceProvider) ->
6  +  helpdeskDashboardState =
7  +    url: "/helpdesk"
8  +    views:
9  +      main:
10 +        controller: "HelpdeskDashboardCtrl"
11 +        templateUrl: "helpdesk/states/dashboard/dashboard.tpl.html"
12 +        data:
13 +          pageTitle: "_Programming Helpdesk_"
14 +          roleWhitelist: ['Student', 'Tutor', 'Convenor', 'Admin']
15 + headerServiceProvider.state 'helpdesk', helpdeskDashboardState
16 +)
17 +.controller("HelpdeskDashboardCtrl", ($scope, $timeout, $rootScope, $interval, $state, HelpdeskStats, HelpdeskTicket, H
18 + # Internal poll interval
19 + pollInterval = null
20 + # How long the duration is between each poll, in seconds
21 + intervalDuration = 30
22 +
23 + #
24 + # Returns true if the stats are already being polled
25 + #
26 + isPolling = -> pollInterval isnt null
27 +
28 + # Keep track of new stats|staff|tickets in these variable
29 + $scope.data =
30 +   tickets: []
31 +   tutorsWorking: []
32 +
33 + #
34 + # This function is called when tickets have been updated
35 + #
36 + ticketsUpdated = (error, tickets) ->
37 +   # Only update what we need to
38 +   resolvedTickets = _.differenceBy($scope.data.tickets, tickets, 'id')
39 +   newTickets = _.differenceBy(tickets, $scope.data.tickets, 'id')
40 +   # Remove resolved tickets + merge new tickets
41 +   $scope.data.tickets =
42 +     _.chain($scope.data.tickets)
43 +     .filter((t) -> !_find(resolvedTickets, { id: t.id })?)
44 +     .concat(newTickets)
45 +     .map((ticket, idx) ->
46 +       # Add ticket's position
47 +       ticket.position = idx + 1
48 +       ticket
49 +     )
50 +     .value()
51 +
52 + #
53 + # This function is called when stats have been updated
54 + #
55 + statsUpdated = (error, stats) ->
56 +   statusColours = {'Quiet': 'primary', 'Available': 'warning', 'BUSY': 'danger'}
57 +   statusEmojis = {'Quiet': '😬', 'Available': '😄', 'BUSY': '😡'}
58 +   statsTimes = _.keys(stats)
59 +   lastRecord = stats[_.last(statsTimes)]
60 +   avgWaitTime = $scope.avgWaitTime = lastRecord.average_wait_time_in_mins
61 +   numUnresolved = $scope.numUnresolved = lastRecord.number_of_unresolved_tickets
62 +   # TODO: Work out the right values
63 +   $scope.averageWaitTimeStatus =
64 +     if avgWaitTime < 6

```

```

65 +     'Quiet'
66 +     else if 6 <= avgWaitTime < 9
67 +     'Available'
68 +     else if avgWaitTime > 9
69 +     'BUSY'
70 +     numberUnresolvedStatus =
71 +     if numUnresolved < 6
72 +     'Quiet'
73 +     else if 6 <= numUnresolved < 9
74 +     'Available'
75 +     else if numUnresolved > 9
76 +     'BUSY'
77 +     $scope.averageWaitTimeEmoji = statusEmojis[$scope.averageWaitTimeStatus]
78 +     $scope.averageWaitTimeColor = statusColours[$scope.averageWaitTimeStatus]
79 +     $scope.numberUnresolvedColor = statusColours[numberUnresolvedStatus]
80 +     # Angular nvd3 accepts milliseconds not seconds for unix time
81 +     $scope.graphData = stats
82 +
83 + #
84 + # This function is called when staff have been updated
85 + #
86 + tutorsUpdated = (error, sessions) ->
87 + # Same concept as ticketsUpdated
88 + sessionsFinished = _.differenceBy($scope.data.tutorsWorking, sessions, 'id')
89 + sessionsStarted = _.differenceBy(sessions, $scope.data.tutorsWorking, 'id')
90 + $scope.data.tutorsWorking =
91 +   _.chain($scope.data.tutorsWorking)
92 +   .filter((s) -> !_.find(sessionsFinished, { id: s.id }))?)
93 +   .concat(sessionsStarted)
94 +   .map((s) ->
95 +     finishIn = s.timeUntilFinish()
96 +     s.minuteDisplay = Math.round(finishIn.minutes() + (finishIn.seconds() * 0.01))
97 +     s.hourDisplay = finishIn.hours()
98 +     if s.minuteDisplay == 60
99 +       s.minuteDisplay = 0
100 +     s.hourDisplay += 1
101 +     s.showHour = s.hourDisplay > 0
102 +     s
103 +   )
104 +   .value()
105 + $scope.helpdeskClosed = $scope.data.tutorsWorking.length is 0
106 + $scope.helpdeskOpen = not $scope.helpdeskClosed
107 +
108 + #
109 + # If clock on event, refresh immediately
110 + #
111 + $rootScope.$on 'CurrentWorkingSession', (event, session) ->
112 +   $scope.pollNow()
113 +
114 + #
115 + # Begins polling for stats
116 + #
117 + startPolling = (interval = intervalDuration) ->
118 +   return if isPolling()
119 +   pollForStats = ->
120 +     # We know if it's the first poll if we are not polling yet!
121 +     isFirstInterval = not isPolling()
122 +     # First poll we want to get the last hour's worth
123 +     subtractValue = if isFirstInterval then 1 * 60 * 60 else interval
124 +     useInterval = if isFirstInterval then null else interval
125 +     from = moment().subtract(subtractValue, 'seconds').format()
126 +     to = moment().format()
127 +     HelpdeskStats.get from, to, useInterval, (error, response) ->

```

```

128 +     $scope.lastUpdated = moment()
129 +     statsUpdated(error, response)
130 +     pollForTickets = ->
131 +     HelpdeskTicket.getUnresolvedTickets null, (error, response) ->
132 +     $scope.lastUpdated = moment()
133 +     ticketsUpdated(error, response)
134 +     pollForTutors = ->
135 +     HelpdeskSession.tutorsWorkingNow (error, response) ->
136 +     $scope.lastUpdated = moment()
137 +     tutorsUpdated(error, response)
138 +     pollFunction = ->
139 +     pollForTickets()
140 +     pollForTutors()
141 +     pollForStats()
142 +     $scope.pollNow = ->
143 +     pollFunction()
144 +     # Call poll at least once to start now
145 +     $scope.pollNow()
146 +     pollInterval = $interval pollFunction, interval * 1000 # milliseconds
147 +
148 +     #
149 +     # Stops helpdesk statistics from polling
150 +     #
151 +     stopPolling = ->
152 +     return unless isPolling()
153 +     $interval.cancel(pollInterval)
154 +     pollInterval = null
155 +
156 +
157 +     #
158 +     # Opens the session modal
159 +     #
160 +     $scope.openHelpdeskSessionModal = HelpdeskSessionModal.show
161 +
162 +     # When we load, start polling at intervalDuration seconds
163 +     startPolling(intervalDuration)
164 +     # When we unload, stop polling
165 +     $scope.$on '$destroy', stopPolling
166 + )

```

105 src/app/helpdesk/states/dashboard/dashboard.scss

View 

```

...
... @@ -0,0 +1,105 @@
1  +// This mixin is used to hide on mobile devices specific rows
2  +@mixin hide-on-mobile {
3  +  @extend .hidden-xs;
4  +  @extend .hidden-sm;
5  +}
6  +
7  +#helpdesk-dashboard {
8  +  // Fix layout for medium
9  +  @media (min-width: $screen-md) {
10 +    height: calc(100vh - #{$navbar-height});
11 +    min-height: 600px;
12 +    display: flex;
13 +    flex-direction: column;
14 +    .row {
15 +      display: flex;
16 +    }
17 +    .status.row {
18 +      flex: 1;
19 +    }
20 +    .graph.row {

```

```
21 +     flex: 2;
22 +   }
23 +   .ticket.row {
24 +     flex: 1.5;
25 +   }
26 + }
27 +
28 + // Hide all rows on mobile except the tickets row
29 + .row:not([class^="ticket"]) {
30 +   @include hide-on-mobile;
31 + }
32 + // Give rows some space between eachother
33 + .row {
34 +   margin: 15px auto 0 auto;
35 + }
36 +
37 + // Help Header
38 + h3.help {
39 +   @extend .text-muted;
40 +   margin-top: 0;
41 +   text-transform: uppercase;
42 +   font-weight: bold;
43 +   font-size: 1.2em;
44 + }
45 +
46 + // Status Row
47 + .status.row {
48 +   font-size: 2em;
49 +   text-align: center;
50 +   .number {
51 +     display: block;
52 +     font-size: 4em;
53 +     line-height: 1;
54 +   }
55 + }
56 + // Tutor panel
57 + .tutors-working {
58 +   align-self: center;
59 +   .list-group {
60 +     max-height: 300px;
61 +     overflow-y: scroll;
62 +   }
63 +   .list-group-item-heading {
64 +     font-size: 1.5em;
65 +     .name {
66 +       font-weight: bold;
67 +     }
68 +   }
69 +   .list-group-item-text {
70 +     margin: 0.7em auto 1em auto;
71 +     font-size: 1.2em;
72 +     display: flex;
73 +     flex-wrap: wrap;
74 +     .label:not(:last-of-type) {
75 +       margin-right: 1ex;
76 +       margin-bottom: 1ex;
77 +     }
78 +   }
79 + }
80 + // Ticket Row
81 + .ticket.row {
82 +   justify-content: center;
83 + }
```

```
84 + // Helpdesk Closed
85 + .helpdesk-closed {
86 +   height: 100%;
87 +   margin-top: 2em;
88 +   display: flex;
89 +   align-items: center;
90 +   text-align: center;
91 +   font-size: 1.5em;
92 +   .message {
93 +     margin: 0 auto;
94 +     button {
95 +       margin-top: 1em;
96 +     }
97 +     .title {
98 +       font-weight: bold;
99 +       display: block;
100 +       margin-top: 1em;
101 +       font-size: 1.5em;
102 +     }
103 +   }
104 + }
105 +}
```

74 src/app/helpdesk/states/dashboard/dashboard.tpl.html

View 

```
... @@ -0,0 +1,74 @@
1 +<div id="helpdesk-dashboard" class="clearfix">
2 +  <div class="helpdesk-closed">
3 +    <div class="message" ng-if="helpdeskClosed">
4 +      <i class='fa fa-hourglass-end fa-4x'></i>
5 +      <span class="title"> Helpdesk Closed </span>
6 +      No tutors are working right now<br>
7 +      <button ng-click="openHelpdeskSessionModal()" if-role="Tutor Convenor Admin" class="btn btn-lg btn-success">
8 +        <i class="fa fa-clock-o"></i> <i class="fa fa-sign-in"></i> Clock On
9 +      </button>
10 +    </div>
11 +    <div class="message" ng-if="helpdeskClosed == null && helpdeskOpen == null">
12 +      <i class="fa fa-spinner fa-pulse fa-4x"></i>
13 +    </div>
14 +  </div><!--/helpdesk-closed-->
15 +  <div ng-if="helpdeskOpen">
16 +    <div class="status row">
17 +      <div class="people-in-queue col-sm-4 text-{{numberUnresolvedColor}}">
18 +        <span class="number">
19 +          {{numUnresolved}}
20 +        </span>
21 +        people in queue
22 +      </div><!--/people-in-queue-->
23 +      <div class="emoji-face col-sm-4">
24 +        <span class="number">{{averageWaitTimeEmoji}}</span> <span> {{averageWaitTimeStatus}}
25 +      </div><!--/emoji-face-->
26 +      <div class="average-wait-time col-sm-4 text-{{averageWaitTimeColor}}">
27 +        <span class="number">
28 +          {{avgWaitTime}}m
29 +        </span>
30 +        average wait time
31 +      </div><!--/average-wait-time-->
32 +    </div><!--/status-row-->
33 +    <div class="graph row">
34 +      <div class="tutors-working col-sm-4">
35 +        <h3 class="help">
36 +          {{ tutorCount = data.tutorsWorking.length }}
37 +          tutor{{tutorCount > 1 ? 's' : ''}} working right now
```

```

38 +     </h3>
39 +     <div class="list-group">
40 +         <div ng-repeat="session in data.tutorsWorking" class="list-group-item">
41 +             <div class="list-group-item-heading">
42 +                 <span class="name">
43 +                     {{finishIn = session.timeUntilFinish(); session.user.name}}
44 +                 </span><!--/tutor-name-->
45 +                 <div class="finish-in pull-right text-muted pointer" tooltip="Finishes at {{session.clock_off_time | date
46 +                     <span ng-if="session.showHour">
47 +                         {{session.hourDisplay}}h
48 +                     </span>
49 +                     {{session.minuteDisplay}}m
50 +                 </div><!--/finish-in-->
51 +             </div>
52 +             <p class="list-group-item-text">
53 +                 <label ng-repeat="unit in session.user.units_taught" class="label label-info pointer" tooltip="{{unit.name
54 +                     {{unit.code}}
55 +                 </label>
56 +             </p><!--/unit-labels-->
57 +         </div>
58 +     </div><!--/tutors-working-list-->
59 + </div><!--/tutors-working-->
60 + <div class="graph col-sm-8">
61 +     <h3 class="help">
62 +         Tickets unresolved and wait times today
63 +     </h3>
64 +     <helpdesk-dashboard-timeline data="graphData" height="400">
65 + </helpdesk-dashboard-timeline>
66 + </div>
67 + </div><!--/graph-row-->
68 + <div class="ticket row">
69 +     <div class="ticket-wrapper col-sm-3" ng-repeat="ticket in data.tickets">
70 +         <helpdesk-ticket ticket="ticket"></helpdesk-ticket>
71 +     </div>
72 + </div><!--/ticket-row-->
73 + </div><!--/helpdesk-open-->
74 +</div>

```

3 src/app/helpdesk/states/states.coffee

View



```

...  ...  @@ -0,0 +1,3 @@
      1  +angular.module('doubtfire.helpdesk.states', [
      2  +  'doubtfire.helpdesk.states.dashboard'
      3  +])

```

2 src/app/tasks/task-definition-selector/task-definition-selector.coffee

View



```

11  11  scope:
12  12      # Unit required
13  13      unit: "="
14  14      + # Model
15  15      + selectedDefinition: '=ngModel'
16  16      # What to do when definition is changed
17  17      onSelectDefinition: "="
18  18      # Provide a btn-style to force the colour to change`

```

96 src/app/visualisations/helpdesk-dashboard-timeline.coffee

View



```

...  ...  @@ -0,0 +1,96 @@
      1  +angular.module('doubtfire.visualisations.helpdesk-dashboard-timeline', [])
      2  +.directive 'helpdeskDashboardTimeline', ->

```

```

3 +   replace: true
4 +   restrict: 'E'
5 +   templateUrl: 'visualisations/visualisation.tpl.html'
6 +   scope:
7 +     rawData: '=data'
8 +     height: '=?'
9 +   controller: ($scope, $timeout, taskService, projectService, Visualisation) ->
10 +     updateData = (rawData) ->
11 +       # Function to fix time as nvd3 needs a unix time in ms, not s
12 +       correctTime = (t) -> t * 1000
13 +       $scope.times = _.chain(rawData).keys().map(correctTime).value()
14 +       $scope.data = [
15 +         {
16 +           key: 'Average Wait Time',
17 +           bar: false,
18 +           values: _.map rawData, (data, time) ->
19 +             {
20 +               x: correctTime(time),
21 +               y: data.average_wait_time_in_mins
22 +             }
23 +         },
24 +         {
25 +           key: 'People In Queue',
26 +           bar: true,
27 +           values: _.map rawData, (data, time) ->
28 +             {
29 +               x: correctTime(time),
30 +               y: data.number_of_unresolved_tickets
31 +             }
32 +         }
33 +       ]
34 +       # Update forceY to be the max num
35 +       barMax =
36 +         _.chain(rawData)
37 +         .map((data) -> data.number_of_unresolved_tickets)
38 +         .max()
39 +       barMax = if barMax > 10 then barMax else 10
40 +       lineMax =
41 +         _.chain(rawData)
42 +         .map((data) -> data.average_wait_time_in_mins)
43 +         .max()
44 +       lineMax = if lineMax > 10 then lineMax else 10
45 +       $scope.options.chart.bars.forceY =
46 +       $scope.options.chart.bars2.forceY =
47 +       [0, barMax]
48 +       $scope.options.chart.lines.forceY =
49 +       $scope.options.chart.lines2.forceY =
50 +       [0, lineMax]
51 +       $timeout ->
52 +         $scope.api.refresh() if $scope.api?.refresh?
53 +
54 +       $scope.$watch (-> _.keys($scope.rawData)[0]), ->
55 +         return unless $scope.rawData?
56 +         updateData $scope.rawData
57 +
58 +       [$scope.options, $scope.config] = Visualisation 'linePlusBarChart', 'Helpdesk Dashboard Timeline', {
59 +         showLabels: no
60 +         margin:
61 +           top: 30,
62 +           right: 30,
63 +           bottom: 50,
64 +           left: 70
65 +         height: $scope.height

```



```

66 +     bars:
67 +         forceY: [0]
68 +     bars2:
69 +         forceY: [0]
70 +     lines:
71 +         forceY: [0]
72 +     lines2:
73 +         forceY: [0]
74 +     color: [
75 +         'darkred',
76 +         '#2ca02c'
77 +     ]
78 +     x: (d, i) -> i
79 +     xAxis:
80 +         axisLabel: 'Time'
81 +         tickFormat: (d) -> moment($scope.times[d]).format('hh:mm a')
82 +     x2Axis:
83 +         showMaxMin: true
84 +         tickFormat: (d) -> moment($scope.times[d]).format('hh:mm a')
85 +     y1Axis:
86 +         axisLabel: 'people waiting'
87 +         tickFormat: (d) -> d
88 +         axisLabelDistance: -12
89 +     y2Axis:
90 +         axisLabel: null
91 +         tickFormat: (d) -> "#{d}m"
92 +     y3Axis:
93 +         tickFormat: (d) -> 'l'
94 +     y4Axis:
95 +         tickFormat: (d) -> 'x'
96 +     }, {}

```

1 src/app/visualisations/visualisations.coffee

View



```

9 9     '@@ -9,6 +9,7 @@ angular.module('doubtfire.visualisations', [
10 10         'doubtfire.visualisations.task-completion-box-plot'
11 11         'doubtfire.visualisations.achievement-box-plot'
12 12         'doubtfire.visualisations.achievement-custom-bar-chart'
12 13         + 'doubtfire.visualisations.helpdesk-dashboard-timeline'
13 14     ])
14 15     .factory('Visualisation', ($interval, analyticsService) ->

```

7 src/styles/common/alerts.scss

View



```

49 49     .alert.alert-warning i.fa:before { content: $fa-var-exclamation-triangle; }
50 50     .alert.alert-success i.fa:before { content: $fa-var-check-circle; }
51 51     }
52 +
53 +// Push down the alert past the header if it has content
54 +@media (min-width: $screen-sm) {
55 +     div[ui-view="header"]:not(:empty) ~ .alert-top {
56 +         top: 66px;
57 +     }
58 +}

```

8 src/styles/common/overrides/modal-overrides.scss

View



```

... 1 // Mobile responsive fixes
... 2 @media (max-width: $screen-md) {

```

```
3 + .modal-footer button {  
4 +   width: 100%;  
5 +   margin-bottom: 1em !important;  
6 +   margin-left: 0 !important;  
7 + }  
8 +}
```