```
var app = angular.module('tournyplanner', ['ngRoute', 'ui.bootstrap',
'angularFileUpload', 'angular-loading-bar']);

// routeProvider routes the user to the right html page and provides the controller
// for that specific page, given the input from the user (buttons etc).
app.config(['$routeProvider', function ($routeProvider) {
  $routeProvider.
    when('/', {
      templateUrl: 'templates/home.html',
      controller: 'HomeController'
    }).
    when('/addTournament', {
      templateUrl: 'templates/addTournament.html',
      controller: 'CreateTournyController'
    }).
    when('/tournament/:tournamentId/edit', {
      templateUrl: 'templates/editTournament.html',
      controller: 'EditTournamentController'
    }).
    when('/tournament/:tournamentId', {
      templateUrl: 'templates/tournament.html',
      controller: 'TournamentController'
    }).
    when('/tournament/:tournamentId/division/:divisionId', {
      templateUrl: 'templates/division.html',
      controller: 'DivisionController'
    }).
    when('/tournament/:tournamentId/division/:divisionId/pool/:poolId', {
      templateUrl: 'templates/pool.html',
      controller: 'PoolController'
    }).
    when('/tournament/:tournamentId/field/:fieldId', {
      templateUrl: 'templates/field.html',
      controller: 'TournamentController'
    }).
    when('/tournament/:tournamentId/division/:divisionId/pool/:poolId/team/:teamId', {
      templateUrl: 'templates/team.html',
      controller: 'TeamDetailController'
    }).
    when('/schedule', {
      templateUrl: 'templates/schedule.html',
      controller: 'ScheduleController'
    }).
    when('/tutorial', {
      templateUrl: 'templates/tutorial.html',
      controller: 'HomeController'
    }).
    otherwise({
      redirectTo: '/'
    });
}]);

app.config(['cfpLoadingBarProvider', function (cfpLoadingBarProvider) {
  cfpLoadingBarProvider.includeSpinner = true;
  cfpLoadingBarProvider.parentSelector = '#navbar';
}]);

app.run(function ($rootScope, $http, $routeParams, $route) {
```

```
  $rootScope.apiUrl = "http://sorenlyng.dk";

  var deleteSchedule = function (tournamentID) {
    console.log("Sletter nuværende kampprogram");
    $http.get($rootScope.apiUrl + "/ScheduleManager/DeleteSchedule?tournamentId=" +
tournamentID)
      .success(function (deleteData) {
        if (deleteData.status === "success")
        {
          console.log("Kampprogram slettet")
          $rootScope.Message = "Kunne ikke gennemfører planlægning. Du kan evt. prøve at
tilføje mere tid eller flere baner.";
        }
        else
        {
          console.log("Kampprogram ikke slettet");
        }
      }).error(function() {

      })
  }

  $rootScope.scheduler = function (tournamentID) {
    console.log("Sletter nuværende kampprogram");
    $rootScope.Message = "Sletter nuværende kampprogram";
    $http.get($rootScope.apiUrl + "/ScheduleManager/DeleteSchedule?tournamentId=" +
tournamentID)
      .success(function (deleteData) {
        console.log("Kampprogram slettet")
        $rootScope.Message = "Kampprogram slettet";
        console.log("Validere turnering");
        $rootScope.Message = "Validere turnering";

        $http.get($rootScope.apiUrl + "/Validator/IsScheduleReady?tournamentID=" +
tournamentID)
          .success(function (validateData) {
            if (validateData.status === "success") {
              console.log("Generer gruppespil");
              $rootScope.Message = "Generer gruppespil";

              $http.get($rootScope.apiUrl +
"/MatchGeneration/GenerateGroupStage?tournamentID=" + tournamentID)
                .success(function (generateGSData) {
                  if (generateGSData.status === "success") {
                    console.log("Generer slutspils hold");
                    $rootScope.Message = "Generer slutspils hold";

                    $http.get($rootScope.apiUrl +
"/MatchGeneration/GenerateFinalsTeams?tournamentID=" + tournamentID)
                      .success(function (generateFTData) {
                        if (generateFTData.status === "success") {
                          console.log("Generer slutspils kampe");
                          $rootScope.Message = "Generer slutspils kampe";

                          $http.get($rootScope.apiUrl +
"/MatchGeneration/GenerateFinalsMatches?tournamentID=" + tournamentID)
                            .success(function (generateFMData) {
                              if (generateFMData.status === "success") {
```

```
                                console.log("Starter planlægning");
                                $rootScope.Message = "Starter planlægning af
kampprogram";
                                $rootScope.Message = "Planlægger kampprogram for eleve
mands kampe";

                                $http.get($rootScope.apiUrl +
"/ScheduleManager/Schedule?tournamentID=" + tournamentID + "&fs=" + 11)
                                  .success(function (eScheduleData) {
                                    if (eScheduleData.status === "success") {
                                      console.log("Eleve mands kampe planlagt");
                                      $rootScope.Message = "Eleve mands kampe planlagt";
                                      $rootScope.Message = "Planlægger kampprogram for
otte mands kampe";

                                      $http.get($rootScope.apiUrl +
"/ScheduleManager/Schedule?tournamentID=" + tournamentID + "&fs=" + 8)
                                        .success(function (eiScheduleData) {
                                          if (eiScheduleData.status === "success") {
                                            console.log("Otte mands kampe planlagt");
                                            $rootScope.Message = "Otte mands kampe
planlagt";
                                            $rootScope.Message = "Planlægger kampprogram
for fem mands kampe";

                                            $http.get($rootScope.apiUrl +
"/ScheduleManager/Schedule?tournamentID=" + tournamentID + "&fs=" + 5)
                                              .success(function (fScheduleData) {
                                                if (fScheduleData.status === "success") {
                                                  console.log("Fem mands kampe
planlagt");
                                                  $rootScope.Message = "Kampprogram
planlagt";
                                                  $http.post($rootScope.apiUrl +
"/Tournament/ChangeIsScheduled", { tournamentId: tournamentID})
                                                  .success(function (data) {
                                                    $rootScope.IsScheduled = true;
                                                    $route.reload();
                                                  })
                                                }
                                                else {
                                                  if(fScheduleData.errorCode === 1 ){
                                                    $rootScope.Message = "Systemets
algoritme fandt ingen løsning";
                                                    $rootScope.Message = "Sletter
kampprogram";
                                                    deleteSchedule(tournamentID);
                                                  }
                                                  else{
                                                    $rootScope.Message = "Fejl ved
planlægningen af fem mands kampe";
                                                    $rootScope.Message = "Sletter
kampprogrammet";
                                                    deleteSchedule(tournamentID);
                                                  }
                                                }
                                              }).error(function () {
                                                if(fScheduleData.errorCode === 1 ){
```

```
                                        $rootScope.Message = "Systemets
algoritme fandt ingen løsning";
                                        $rootScope.Message = "Sletter
kampprogram";
                                        deleteSchedule(tournamentID);
                                      }
                                      else{
                                        $rootScope.Message = "Fejl ved
planlægningen af fem mands kampe";
                                        $rootScope.Message = "Sletter
kampprogrammet";
                                        deleteSchedule(tournamentID);
                                      }
                                    })
                                  }
                                  else {
                                    if(eiScheduleData.errorCode === 1){
                                      $rootScope.Message = "Systemets algoritme
fandt ingen løsning";
                                      $rootScope.Message = "Sletter kampprogram";
                                      deleteSchedule(tournamentID);
                                    }
                                    else {
                                      $rootScope.Message = "Fejl ved
planlægningen af otte mands kampe";
                                      $rootScope.Message = "Sletter kampprogram";
                                      deleteSchedule(tournamentID);
                                    }
                                  }
                                }).error(function () {
                                  if(eiScheduleData.errorCode === 1){
                                    $rootScope.Message = "Systemets algoritme
fandt ingen løsning";
                                    $rootScope.Message = "Sletter kampprogram";
                                    deleteSchedule(tournamentID);
                                  }
                                  else {
                                    $rootScope.Message = "Fejl ved planlægningen
af otte mands kampe";
                                    $rootScope.Message = "Sletter kampprogram";
                                    deleteSchedule(tournamentID);
                                  }
                                })
                              }
                            else {
                              if(eScheduleData.errorCode === 1) {
                                $rootScope.Message = "Systemets algoritme fandt
ingen løsning";
                                $rootScope.Message = "Sletter kampprogram";
                                deleteSchedule(tournamentID);
                              }
                              else {
                                $rootScope.Message = "Fejl ved planlægningen af
eleve mands kampe";
                                $rootScope.Message = "Sletter kampprogram";
                                deleteSchedule(tournamentID);
                              }
                            }
```

```
                                    }).error(function () {
                                      if(eScheduleData.errorCode === 1) {
                                        $rootScope.Message = "Systemets algoritme fandt
ingen løsning";
                                        $rootScope.Message = "Sletter kampprogram";
                                        deleteSchedule(tournamentID);
                                      }
                                      else {
                                        $rootScope.Message = "Fejl ved planlægningen af
eleve mands kampe";
                                        $rootScope.Message = "Sletter kampprogram";
                                        deleteSchedule(tournamentID);
                                      }
                                    })
                                  }
                                  else {
                                    $rootScope.Message = "Fejl ved genereringen af slutspils
kampene";
                                    $rootScope.Message = "Sletter kampprogram";
                                    deleteSchedule(tournamentID);
                                  }
                                }).error(function () {
                                  $rootScope.Message = "Fejl ved genereringen af slutspils
kampene";
                                  $rootScope.Message = "Sletter kampprogram";
                                  deleteSchedule(tournamentID);
                                })
                              }
                              else {
                                $rootScope.Message = "Fejl ved genereringen af slutspils
holdene";
                                $rootScope.Message = "Sletter kampprogram";
                                deleteSchedule(tournamentID);
                              }
                            }).error(function () {
                              $rootScope.Message = "Fejl ved genereringen af slutspils
holdene";
                              $rootScope.Message = "Sletter kampprogram";
                              deleteSchedule(tournamentID);
                            })
                          }
                          else {
                            $rootScope.Message = "Fejl ved genereringen af gruppespillet";
                            $rootScope.Message = "Sletter kampprogram";
                            deleteSchedule(tournamentID);
                          }
                        }).error(function () {
                          $rootScope.Message = "Fejl ved genereringen af gruppespillet";
                          $rootScope.Message = "Sletter kampprogram";
                          deleteSchedule(tournamentID);
                        })
                      }
                      else {
                        $rootScope.Message = "Fejl ved valideringen af turneringen";
                        $rootScope.Message = "Sletter kampprogram";
                        deleteSchedule(tournamentID);
                      }
                    }).error(function () {
```

```
            $rootScope.Message = "Fejl ved valideringen af turneringen";
            $rootScope.Message = "Sletter kampprogram";
            deleteSchedule(tournamentID);
          })
        if(deleteData.status === "success")
        {
          console.log("Kampprogram slettet")
        }
        else
        {
          console.log("Fejl ved sletningen af kampprogrammet")
        }
      }).error(function () {

      })
    }
});

// HomeController is the controller for the home.html page,
// where the "log-in" or "create new tournament" options are available.
app.controller('HomeController', ['$scope', '$http', '$location', '$rootScope', function
($scope, $http, $location, $rootScope) {

  $scope.password = "";

  $scope.errMsg = function () {
    $scope.error = !$scope.error;
  }

  // getId is the "log-in" to a tournament, which needs the password parameter,
  // to redirect the user to the specific tournament.
  $scope.getId = function (password) {
    $http.post($rootScope.apiUrl + "/Tournament/IdFromPass", { password: password })
      .success(function (passwordData) {
        if (passwordData.Id != 0) {
          $scope.error = false;
          $location.path("tournament/" + passwordData.Id);
        }
        else {
          $scope.error = true;
        }
      }).error(function (err) {
        $scope.error = err;
      });
  }
}]);

app.filter('jsonDate', ['$filter', function ($filter) {
  return function (input) {
    return (input) ? $filter('date')(parseInt(input.substr(6)), "yyyy-MM-dd HH:mm") : '';
  };
}]);
app.filter('jsonOnlyDate', ['$filter', function ($filter) {
  return function (input) {
    return (input) ? $filter('date')(parseInt(input.substr(6)), "yyyy-MM-dd") : '';
  };
}]);
app.filter('jsonOnlyTime', ['$filter', function ($filter) {
```

```
  return function (input) {
    return (input) ? $filter('date')(parseInt(input.substr(6)), "HH:mm") : '';
  };
}]);
app.filter('jsonOnlyHour', ['$filter', function ($filter) {
  return function (input) {
    return (input) ? $filter('date')(parseInt(input.substr(6)), "HH") : '';
  };
}]);
```