

Exercise (Instructions): Ionic Modals and Forms

Objectives and Outcomes

In this exercise you will explore how Ionic modals are constructed and used to display information. In addition you will learn about Ionic forms support. At the end of this exercise, you will be able to:

- Construct, show and hide modals within your app to display information
- Design forms using Ionic's support for form elements.

Exploring Modals and Forms in Ionic

The conFusion app that we have scaffolded out contains a login modal already. You will examine this app to see the design of the login modal and the form included within the login modal. Constructing a modal requires two parts, (1) defining the template for the modal, and (2) controlling the modal from code.

- Examine the login modal template that is defined in *login.html* file. See how the form for soliciting the username and password are designed in the template.
- Open *controllers.js* file and see how the login modal is activated in the *AppCtrl* controller. In particular, note the use of the *\$ionicModal* and *\$timeout* service, and the functions provided by the service to show and hide the modal.

Creating a Reservation Form

You will now create a modal with a reservation form that will enable users to reserve a table at the restaurant.

- Add a new item into the sidebar that will enable the user to show the modal containing the reservation form by adding the following code to *sidebar.html*:

```
<ion-item menu-close ng-click="reserve()">
  Reserve Table
</ion-item>
```

Creating reserve.html Modal Template

- Create a new file in *templates* folder named *reserve.html*, and add the following code to create the reservation form:

```

<ion-modal-view>
  <ion-header-bar>
    <h1 class="title">Reserve Table</h1>
    <div class="buttons">
      <button class="button button-clear" ng-click="closeReserve()">Close</b
utton>
    </div>
  </ion-header-bar>
  <ion-content>
    <form id="reserveForm" name="reserveForm" ng-submit="doReserve()">
      <div class="list">
        <label class="item item-input item-select">
          <span class="input-label">Number of Guests</span>
          <select ng-model="reservation.numGuests">
            <option ng-repeat="n in [1,2,3,4,5,6]" value="{{n}}">{{n}}</
option>
          </select>
        </label>
        <li class="item item-toggle">
          Smoking?
          <label class="toggle toggle-assertive">
            <input type="checkbox" ng-model="reservation.smoking">
            <div class="track">
              <div class="handle">
            </div>
          </div>
        </label>
        </li>
        <label class="item item-input">
          <span class="input-label">Date</span>
          <input type="date" ng-model="reservation.date">
        </label>
        <label class="item item-input">
          <span class="input-label">Time</span>
          <input type="time" ng-model="reservation.time"
            name="reservetime" min="11:00:00" max="21:00:00">
        </label>
        <label class="item">
          <button class="button button-block button-positive" type="submit">
Reserve</button>
        </label>
      </div>
    </form>
  </ion-content>
</ion-modal-view>

```

Activating the Modal in the Controller

- In *controllers.js* file in the AppCtrl controller, add the following code to activate the modal:

```
// Create the reserve modal that we will use later
$ionicModal.fromTemplateUrl('templates/reserve.html', {
  scope: $scope
}).then(function(modal) {
  $scope.reserveform = modal;
});

// Triggered in the reserve modal to close it
$scope.closeReserve = function() {
  $scope.reserveform.hide();
};

// Open the reserve modal
$scope.reserve = function() {
  $scope.reserveform.show();
};

// Perform the reserve action when the user submits the reserve form
$scope.doReserve = function() {
  console.log('Doing reservation', $scope.reservation);

  // Simulate a reservation delay. Remove this and replace with your reservation
  // code if using a server system
  $timeout(function() {
    $scope.closeReserve();
  }, 1000);
};
```

- Save the changes and have a look at the resulting application.

Conclusions

In this exercise you explored the Ionic support for modals and forms in detail.

