Exception Handling

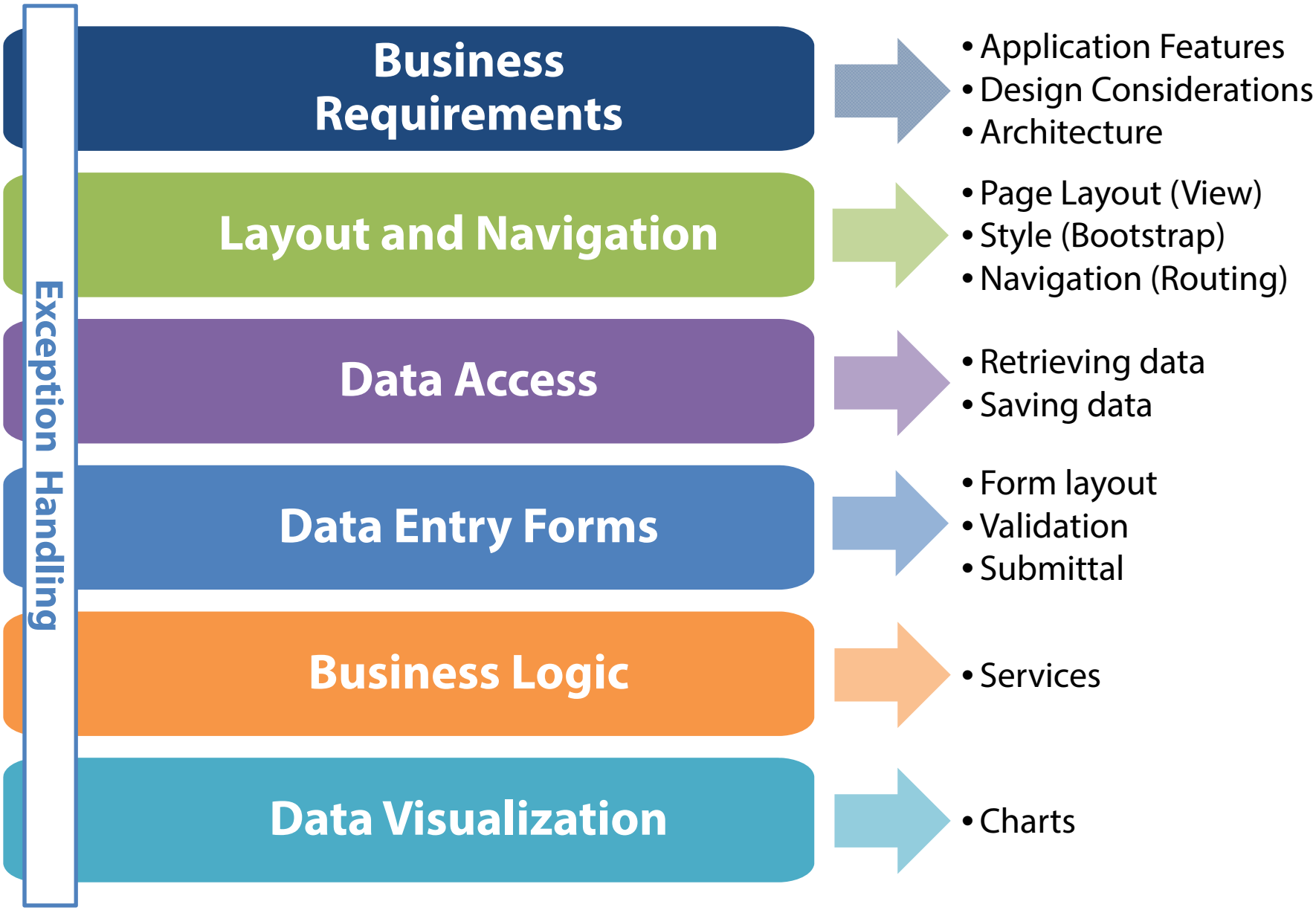
Deborah Kurata
http://blogs.msmvps.com/deborahk/
@DeborahKurata

pluralsight
hardcore dev and IT training

Module Objectives

- Evaluate Exception Handling Options
- Catch runtime errors with try/catch
- Use the global exception handler
- Implement failure functions

Angular Line of Business Applications



Exception Handling

Prevent the error

Catch the exception

Notify the user

Log

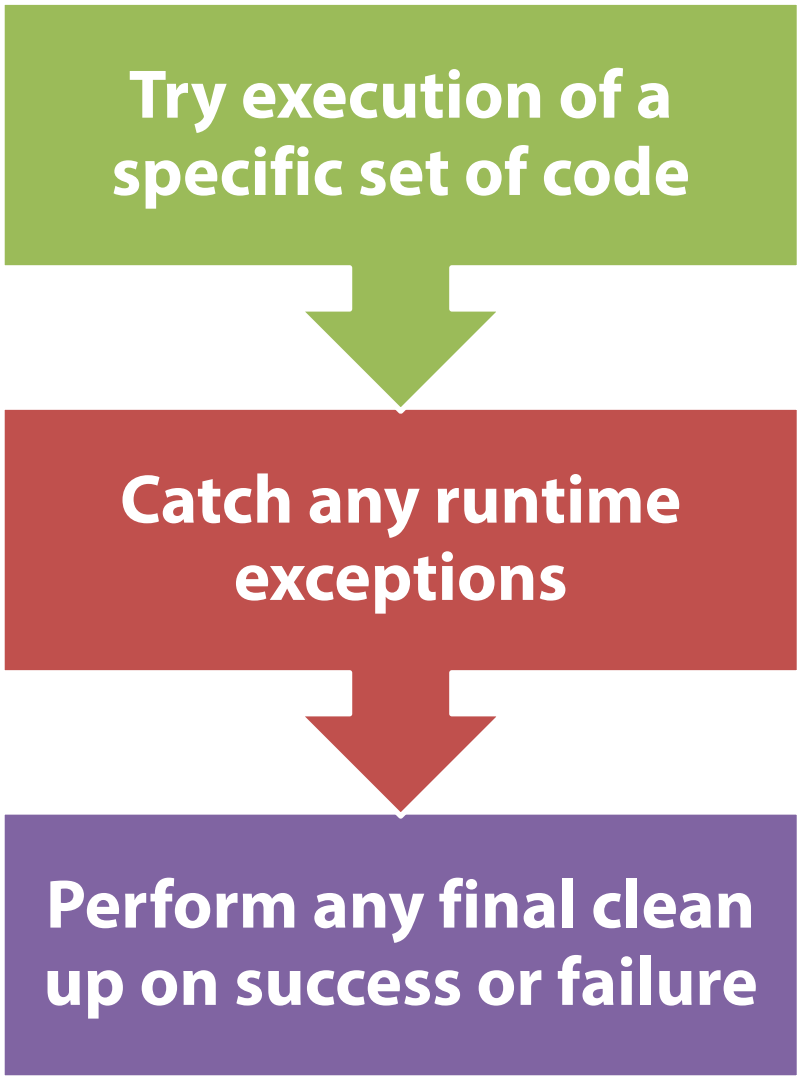
Catching Errors

try/catch Block

Failure functions

Global exception handler: $exceptionHandler

try/catch Blocks




```
vm.addTags = function (tags) {
  try {
   var array = tags.split(',');
   vm.tags =
         vm.tags.concat(array);
   vm.newTags = "";
  } catch (e) {
   alert("Please enter tags")
  }
}
vm.addTags = function (tags) {
  if (tags) {
   var array = tags.split(',');
   vm.tags =
         vm.tags.concat(array);
   vm.newTags = "";
  } else {
   alert("Please enter tags")
  }
}
```

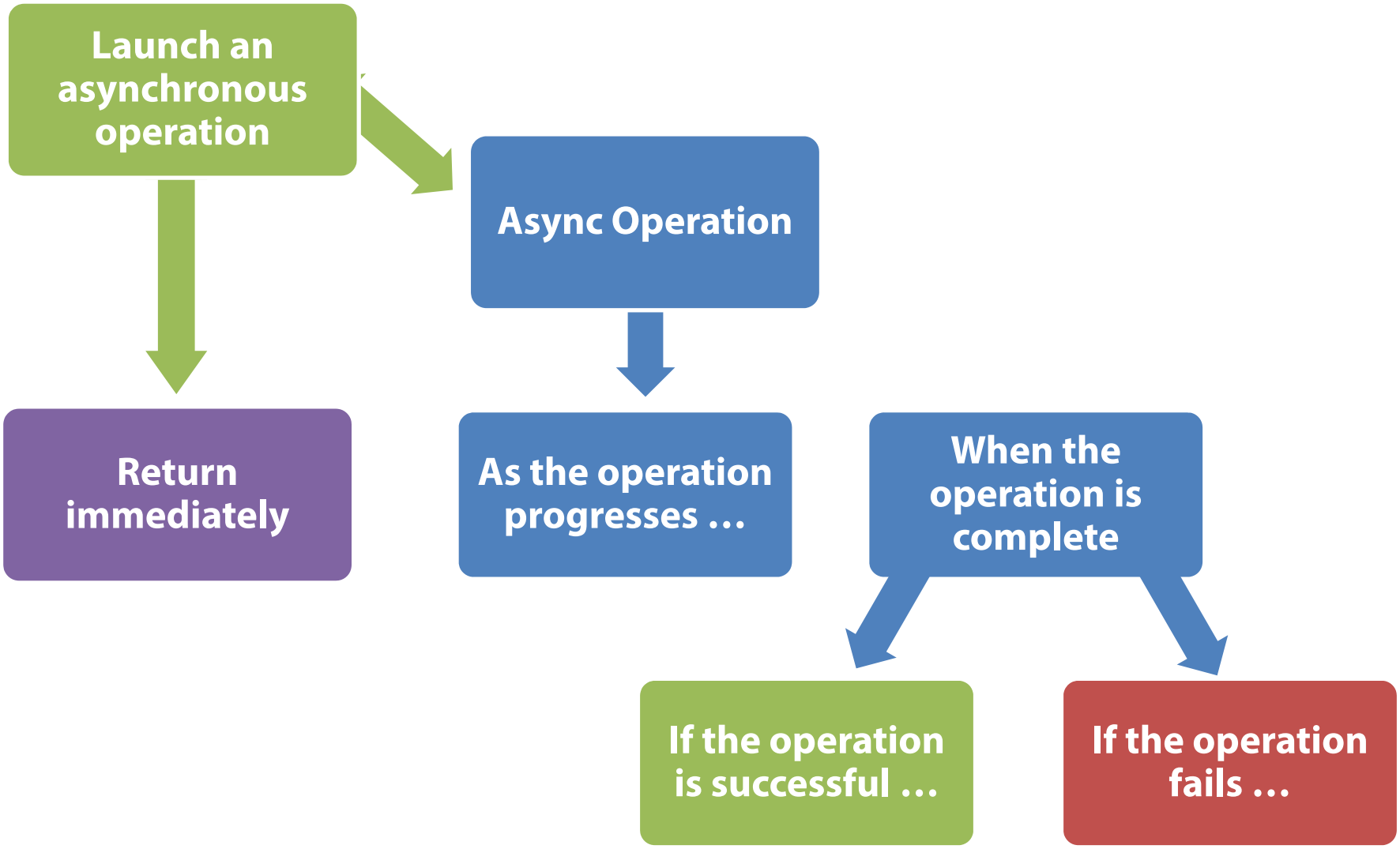
$exceptionHandler Service

- Built-in Angular service
- Any uncaught exception is delegated to this service
- Only catches exceptions
- Default implementation logs the error to the browser console
- You can decorate this service to customize it

Decorating $exceptionHandler

```
app.config(function ($provide) {
  $provide.decorator("$exceptionHandler",
                     ["$delegate",
   function ($delegate) {
    return function (exception, cause) {
     exception.message = "Please contact the Help Desk!
                     \n Message: " + exception.message;
       $delegate(exception, cause);
       alert(exception.message);
     };
   }]);
});
```

Angular Promise



Failure Callback Function

Angular promises provide a then function

- Executed when the promise is resolved or rejected

then takes three parameters

- Success callback function
- Failure callback function
- Notification callback function

```
somePromise.
   then(function(response)
          { alert('Success'); },
        function(response)
          { alert('Failed'); },
        function(statusNotification)
         { alert('Got notification'); }
);
```

Failure Function and $resource

```
products: function (productResource) {
 return productResource.query().$promise;
}

products: function (productResource) {
  return productResource.query(
    function(response) {
       // no code needed for success
    },
    function(response) {
       alert(response.statusText);
    }
  ).$promise;
}
```

Module Objectives