Themenliste Exam 70-480

# Implement and manipulate document structures and objects (24%)

## Create the document structure ☺

Beschreibung:

Structure the UI by using semantic markup, including for search engines and screen readers (Section, Article, Nav, Header, Footer, and Aside); create a layout container in HTML

### Was könnte das bedeuten:

* Die Grundstrukturelemente seit HTML 5 die einige der div-Elemente reduzieren sollen (die oben genannten)

### Status: DONE ☺

Aber vertieft sollte es trotzdem werden. Die einzelnen Elemente sollte man aus dem Effeff beherschen.

## Write code that interacts with UI controls ☹

### Beschreibung

Programmatically add and modify HTML elements; implement media controls; implement HTML5 canvas and SVG graphics

### Was könnte das bedeuten:

* Javascript um HTML DOM Elemente hinzuzufügen und manipulieren (**done**)
* Media Controlls aus HTML5 (audio, video)
* Canvas und SVG

### Status

DOM wichtig und schon einigermaßen verinnerlicht. Der Rest für später aufheben, da in täglicher Arbeit keine Verwendung.

## Apply styling to HTML elements programmatically ☹

### Beschreibung:

Change the location of an element; apply a transform; show and hide elements

### Was könnte das bedeuten

* Change Location:
* show and hide: e.style.display = 'none'; oder ‘block’

Javascript, aber nicht ganz klar was genau, zumindestens viel dom und CSS background

## Implement HTML5 APIs 😐

### Beschreibung

Implement storage APIs, AppCache API, and Geolocation API

### Was könnte das Bedeuten

* HTML5 Web Storage einfaches key/Value mit localStorage.key = Value für Daten ohne Zeitbeschränkung und sessionStorage, welcher nach Beendigung der Session gelöscht wird (Frage: Was hat das mit HTML5 zu tun, ist das nicht Javascript)
* AppCache: Dateien die gecached werden, um auch einen offline-Modus anbieten zu können (mittels Fallback)
* Geolocation: Abfragen der Standortinfos per Javascript und eventuel ne Karte von google runterladen, die dort zentriet ist

### Status

Nicht mehr wahnsinnig viel zu machen, ein bisschen üben.

## Establish the scope of objects and variables 😐

### Beschreibung

Define the lifetime of variables; keep objects out of the global namespace; use the “this” keyword to reference an object that fired an event; scope variables locally and globally

### Was könnte das Bedeuten

Umgang mit Variablen, Objekten, Namespaces und Scopes in Javascript

### Status

Schon mal was davon gehört, ein bisschen Praxis notwendig

## Create and implement objects and methods 😐

### Beschreibung

Implement native objects; create custom objects and custom properties for native objects using prototypes and functions; inherit from an object; implement native methods and create custom methods

### Was könnte das Bedeuten

Standard Umgang mit Javascript und dessen „Objectorientierung“

### Status

Fortgeschritten, aber mehr Übung notwendig

# Implement Program Flow (25%)

## Allgemein

* Grundlegender Javascript Kontrollfluss noch relativ einfach
* Ab Onblur und Co. wird happig

## Implement program flow ☺

### Beschreibung

Iterate across collections and array items; manage program decisions by using switch statements, if/then, and operators; evaluate expressions

### Was könnte das bedeuten

* JavaScript Grundlagen, Verständnis der Syntax, Arrays (und Collections) noch das komplizierteste
* If/then, switch und for..in Statements eigentlich recht unkompliziert
* eval(..) erwartet eine string, der als normalen javascript code ausgeführt wird, damit könnte man dynamisch javascript code erzeugen und ausführen, ansonsten nichts dramatisches

### Status: Done

Wie immer üben, Syntax ist aber recht vertraut

## Raise and handle an event 😐

### Beschreibung

Handle common events exposed by DOM (OnBlur, OnFocus, OnClick); declare and handle bubbled events; handle an event by using an anonymous function

### Was könnte das bedeuten

* DOM Events: OnBlur,OnFocus,Onclick … ☺
* bubble events: unhandled events zum parent Element weitergeben ☺
* handle an event by using an anonymous function: TODO, aber kaum supported

## Implement exception handling ☺

### Beschreibung

Set and respond to error codes; throw an exception; request for null checks; implement try-catch-finally blocks

## Implement a callback ☹

### Beschreibung

Receive messages from the HTML5 WebSocket API; use jQuery to make an AJAX call; wire up an event; implement a callback by using anonymous functions; handle the “this” pointer

### Was könnte das bedeuten

Hardcore WebSocket, AJAX stuff

## Create a web worker process ☺

### Beschreibung

Start and stop a web worker; pass data to a web worker; configure timeouts and intervals on the web worker; register an event listener for the web worker; limitations of a web worker

### Status

Eigentlich recht einfach, siehe Notizen

# Access and secure data (26%)

## Validate user input by using HTML5 elements

### Beschreibung

Choose the appropriate controls based on requirements; implement HTML input types and content attributes (for example, required) to collect user input

### Was könnte das bedeuten

Siehe Video

## Validate user input by using JavaScript

### Beschreibung

Evaluate a regular expression to validate the input format; validate that you are getting the right kind of data type by using built-in functions; prevent code injection

# Allgemeine DOTO

* Tabbellen
* Mit JavaScript (JQuery) Elemente verschieben
* Validation (aber wichtig für Arbeiten)
* "Programmatically add and modify HTML elements; implement media controls; implement HTML5 canvas and SVG graphics
  + DOM mitlerweile klar, auch JQuery
  + HTML elemente hinzufügen (mit JQuery), noch üben
  + HTML5 canvas uns SVG offen
* "Implement storage APIs, AppCache API, and Geolocation API"
  + storage verinnerlichen
  + Rest Unterlagen prüfen
* Javascript Gültigkeitsbereiche (global, local, this)
* "bubbled events"
  + wann ist ein Event "gebubbled"
  + sind gefangene Events automatisch aus den "bubble" Vorgang entfernt
* das "null" Objekt (JS)
* AJAX Calls verinnerlichen
* Implementierung von WebWorkern (Notizen), Timeouts
* (De)Serialisierung von Daten (JSON, XML)
* Text-Styling
* CSS Box-Properties (runde Ecken ...)
* Prototyping