Hier nochmal die Themen/der Ablauf:

* Wiederholung – Matthias
  + HTML/CSS
* Webseitenidee/sinn präsentieren – Olli
  + Passwortstärke testen [Entropie recherchieren]
  + Dynamische Inhalte:
    - Bsp Uhrzeit
* Was ist JavaScript? – Matthias **5 Minuten**
  + File einbetten
* Hello World Programm – Matthias: **5/30 Minuten**
  + ~~Button~~ clicken = alert
  + Direkt im HTML
* DOM Manipulation – Olli: **10/30 Minuten**
  + ~~🡺 Events (zb mit keyup) für~~ ~~Passwortstärke~~
  + Button onclick anpassen
  + CSS Selectors
    - ~~„select“ function bauen~~
  + Events
    - ~~Content in Divs aufgeteilt~~
* Kontrollstrukturen – Matthias: **5/30 Minuten**
  + ~~if/else~~/case/~~Schleifen~~ (auch: truthy)
    - Truthy: Präsentation
    - DestroyAllSoftware talk
  + ~~Funktionen~~
    - ~~Bubble Sort~~
* Rekursion – Olli: **5/30 Minuten**
  + ~~Fibonacci Zahlen~~
  + ~~Binäre Suche~~
* Objektorientiert – Matthias: **15/45 Minuten**
  + ~~Über Funktionen~~
  + ~~Object literal~~
  + Prototypen
    - Klasse „Auto“, Dropdown 🡺 Infos anzeigen
* Funktionales Programmieren – **Olli: 15/45 Minuten**
  + An Auto Objekt anknüpfen
    - Neue Funktion doSchritt, ruft sich selbst pro Sekunde auf
    - ~~Bubble Sort ändern:~~ 
      * ~~Dann am Ende callback von doSchritt~~
  + Funktionen als Objekte
  + Anonyme Funktionen
  + 🡺 Callbacks
    - Sort(algorithm, array, callback)
      * ~~SelectionSort (n^2)~~
      * ~~MergeSort (n log n)~~
      * ~~Quicksort (n log n)~~
      * ~~Insertionsort n^2~~
    - Visualize them
    - ~~For loop funktional~~
* Scope und Context – Olli: **10/30 Minuten**
  + Alles bisherig programmiertes nicht mehr in Globalen Scope tun
    - ~~Einlesen, unterschied scope/context~~
* Falls noch Zeit ist:
  + AJAX – Matthias: **10/30 Minuten**
    - http://jscc.herokuapp.com
    - Einleitung:
      * GET ping,
      * POST echo,
      * GET fehler (400), not found (404)
      * ~~Serverseitig~~
    - Asynchronität
    - app.get("/ping", function(req, res){
    - res.send("Hello World!");
    - });
    - app.post("/echo", function(req, res) {
    - res.send(req.body.message);
    - });
    - app.get("/error", function(req, res) {
    - res.status(400).end();
    - });
    - app.get("/forbidden", function(req, res) {
    - res.status(403).end();
    - })
    - app.get("/notFound", function(req, res) {
    - res.status(404).end();
    - });
    - (JSON)
  + *Programmierparadigma:* ***15/30 Minuten***
    - *Globale Objekte, nicht alles ins global scope*
    - *Fluent interface*
    - *Module*
  + WebSockets – **Olli: 5/30 Minuten**
    - Chat
  + *WebWorker*
* Ressourcenwebseite
  + Google
  + W3Schools
  + SelfHTML
  + CSS4you
  + Fortgeschritten: http://addyosmani.com/resources/essentialjsdesignpatterns/book/

Olli:

* ~~Dropdown ins HTML,~~
* ~~bessere Uhrzeitdarstellung,~~
* ~~bessere timeToCrack Darstelljung~~
* ~~Mehrere Content Divs, Tabs display oder nicht display über CSS Klassen~~
* ~~Rekursion ins HTML reinbringen~~
* ~~Heroku Server Backend organisieren~~
* Visualisieren von Sortieralgorithmen
* Backend chat entwickeln

Todo:

* Präsentation
  + Jeder seinen Teil: Wie lange Präsentation, wie lange Zeit zum üben?
  + Dokumentation der Teile im README Format in Github