

## 2η Σειρά Ασκήσεων

Διάρκεια: 28/10 – 12/11  
Αξία: 10% του τελικού σας βαθμού  
Θεματική ενότητα : **JavaScript - APIs**

### Άσκηση 1. Liquid Democracy [70%]

**Αποτελεί μέρος του συνόλου των ασκήσεων που θα συνενωθούν στο τέλος.**

Εμπλουτισμός/βελτίωση της φόρμας εγγραφής που φτιάξατε στην άσκηση 1.

#### I. (35%) DOM manipulation - google maps and geocode

##### **a) (5%) Password Mismatch - DOM manipulation**

Θα πρέπει να ελεγχθεί ότι τα 2 passwords που έχουν εισαχθεί από τον χρήστη είναι όντως τα ίδια. Σε περίπτωση λάθους θα πρέπει να τυπωθεί κατάλληλο μήνυμα δίπλα από τα πεδία (όχι με χρήση alert).

##### **b) (15%) Χρήση google geocode API - επαλήθευση πόλης**

Καλείστε να εισάγετε ένα προαιρετικό (για την εγγραφή ενός χρήστη) πεδίο με το όνομα "Διεύθυνση" στη φόρμα στο οποίο θα μπορεί να εισάγει ο χρήστης τη διεύθυνσή του. Στη συνέχεια συνενώνοντας τις τιμές των πεδίων "Χώρα", "Πόλη" και "Διεύθυνση", θα πρέπει να επαληθεύσετε μέσω του geocode service αν η συγκεκριμένη περιοχή είναι αποδεκτή από την υπηρεσία ή όχι (θεωρήστε ότι η τοποθεσία υπάρχει εφόσον το API επιστρέφει μία μη κενή λίστα αποτελεσμάτων). Σε περίπτωση που η απάντηση του service είναι κενή (άρα δηλαδή δεν γνωρίζει το google τη συγκεκριμένη τοποθεσία), θα πρέπει να τυπώνεται κατάλληλο μήνυμα λάθους στη σελίδα, σε κάποιο dom element (όχι χρήση alert).

##### **c) (15%) Χρήση google maps - εμφάνιση χάρτη**

Δεδομένης μιας σωστής τοποθεσίας (Χώρα, Πόλη και Διεύθυνση) θα πρέπει να δίνεται η επιλογή στο χρήστη (μέσω κάποιου κουμπιού) να βλέπει ένα χάρτη με ένα marker πάνω στη πόλη που δήλωσε ο χρήστης. Σε περίπτωση που ο χρήστης αλλάξει την πόλη, θα πρέπει να χαθούν από το DOM τα elements που σχετίζονται με το button και τον container (π.χ. div) του χάρτη. Σε περίπτωση που το geocode έχει επιστρέψει περισσότερες από μία σωστές τοποθεσίες, μπορείτε να χρησιμοποιήσετε την πρώτη.

#### **Χρήση google maps - geocoding services**

Καλείστε να χρησιμοποιήσετε το API των google maps όσο και το geocoding service. Για τη χρήση του google maps θα χρειαστεί να ζητήσετε ένα API Key απ' τη google όπως περιγράφεται στο δεύτερο από τα links που παρατίθενται. Βοηθητικοί σύνδεσμοι:

- <https://developers.google.com/maps/documentation/geocoding/intro>
- <https://google-developers.appspot.com/maps/documentation/javascript/adding-a-google-map>
- <https://google-developers.appspot.com/maps/documentation/javascript/examples/geocoding-simple>

Σημείωση: Σημαντικό είναι να καταλάβετε τι ακριβώς κάνει ο κώδικας και να αποφύγετε απλά το έτοιμο copy-paste (θυμηθείτε τα αντικείμενα που είδαμε στο μάθημα). Προσπαθήστε οι συναρτήσεις που θα φτιάξετε να είναι γενικές έτσι ώστε π.χ. αν στο μέλλον χρειαστεί να ζωγραφίσετε το map σε κάποιο άλλο div να μπορείτε να το κάνετε εύκολα δίνοντας το κατάλληλο όρισμα στην αντίστοιχη συνάρτηση. Αντίστοιχα αν χρειαστεί να εισάγετε νέους markers να μπορεί να γίνει μέσω της κατάλληλης κλήσης μιας συνάρτησης, η οποία θα μπορεί να δέχεται το map πάνω στο οποίο θέλουμε να ζωγραφίσουμε τους markers και το αντίστοιχο location.

## **II (35%) Χρήση εξωτερικών services – Αναγνώριση Προσώπων – Ajax Requests**

Στη συγκεκριμένη άσκηση θα αξιοποιήσουμε την online υπηρεσία faceplusplus η οποία μας δίνει τη δυνατότητα για face detection και face recognition σε εικόνες, έτσι ώστε μέσω αναγνώρισης προσώπου να συμπληρώνεται αυτόματα το username. Για τη συγκεκριμένη άσκηση σας έχει δοθεί κώδικας τον οποίο καλείστε να χρησιμοποιήσετε και ο οποίος είναι διαθέσιμος στο repository [https://bitbucket.org/papadako/forstudents\\_hy359\\_a2\\_2017/](https://bitbucket.org/papadako/forstudents_hy359_a2_2017/). Ο κώδικας περιέχει ένα html αρχείο το οποίο κάνει χρήση των media δυνατοτήτων των browsers ώστε να εμφανίζει το video που καταγράφει η camera του υπολογιστή σας. Επιπλέον υπάρχει ένα javascript αρχείο με βασική λειτουργικότητα που θα χρειαστείτε (π.χ. να τραβήξετε μία φωτογραφία), και το οποίο θα πρέπει να εμπλουτίσετε (το αρχείο έχει επισημειωθεί με σχόλια για να σας καθοδηγήσουν).

### **α) (25%) Upload φωτογραφίας και αναγνώριση ύπαρξης προσώπου κατά το registration - Αποθήκευση της φωτογραφίας online με συγκεκριμένο id (username)**

Σας ζητείται να επεκτείνετε την φόρμα εγγραφής αξιοποιώντας το κώδικα που σας έχει δοθεί έτσι ώστε να μπορεί κάποιος προαιρετικά κατά την εγγραφή του να συσχετίσει το λογαριασμό που δημιουργεί (username) με μία συγκεκριμένη φωτογραφία που θα τραβήξει μέσω της κάμερας. Συγκεκριμένα θα πρέπει να υπάρχει ένα radio button το οποίο δεδομένου ότι έχει εισαχθεί ένα username από τον χρήστη θα ενεργοποιεί το video και τα κατάλληλα buttons "Take photo" και "Upload" (δείτε τον κώδικα που σας δόθηκε). Μέσω του upload ο χρήστης θα πρέπει να μπορεί να στείλει τη φωτογραφία στον server ώστε να κάνει την αναγνώριση ύπαρξης κάποιου προσώπου. Σε περίπτωση επιτυχημένης αναγνώρισης το service θα επιστρέφει ένα JSON με τα μοναδικά face\_tokens (ids) για κάθε πρόσωπο που αναγνώρισε. Σε αυτή τη περίπτωση θα πρέπει το συγκεκριμένο face\_token να συσχετιστεί με το αντίστοιχο username που έχει δοθεί από το χρήστη και να σωθεί στο κατάλληλο set (faceSet) από εικόνες που έχει φτιαχτεί για το μάθημα με το όνομα hy359. Για να το επιτύχετε αυτό θα πρέπει να κάνετε τα κατάλληλα ajax requests στο service, τα οποία και καλείστε να υλοποιήσετε (οδηγίες υπάρχουν στο javascript αρχείο που σας έχει δοθεί - δείτε και το αντικείμενο faceAPI).

Συγκεκριμένα σας ενδιαφέρουν τα εξής requests, όπου τα 2 τελευταία προϋποθέτουν ότι το πρώτο ήταν επιτυχές και ότι έγινε αναγνώριση ενός προσώπου όπου του δόθηκε και ένα μοναδικό face\_token (μοναδικό αναγνωριστικό στη πλευρά του server).

1. Detect (<https://console.faceplusplus.com/documents/5679127>) (δείτε τα requests parameters καθώς και το τι επιστρέφεται στο response σε περίπτωση αποτυχίας ή επιτυχίας)
2. Εφόσον έχει αναγνωριστεί ένα πρόσωπο (έχετε ένα face\_token) μπορείτε να του δώσετε ένα αναγνωριστικό (το username στη περίπτωσή μας). Για να το επιτύχετε αυτό μπορείτε να χρησιμοποιήσετε το Face SetUserID service (<https://console.faceplusplus.com/documents/6329500>). Προαιρετικά, αν θέλετε να δείτε ότι όλα πήγαν καλά μπορείτε να χρησιμοποιήσετε την υπηρεσία Face GetDetail (<https://console.faceplusplus.com/documents/6329496>).
3. Εφόσον έχει αναγνωριστεί ένα πρόσωπο (έχετε ένα face\_token) και του έχετε δώσει και το επιθυμητό username, μπορείτε να το σώσετε στον online server (διαφορετικά θα χαθεί σε 72 ώρες!). Για το λόγο αυτό θα χρησιμοποιήσετε την υπηρεσία FaceSet:AddFace (<https://console.faceplusplus.com/documents/6329371>) ώστε να προσθέσετε το face\_token που σας επιστράφηκε από το Detect στο set από faces που έχει φτιαχτεί για το μάθημα με outer\_id:hy359.

Για τα requests μπορείτε να συμπληρώσετε τα παρακάτω credentials στο faceAPI αντικείμενο που σας έχει δοθεί (line 15).

```
apiKey: 'l2jNgKbkIHXSr4vMzNygHXx2g8c_xT9c',  
apiSecret: '2T6XdZt4EYw-l7OhmZ6glwtECI8le_lp',
```

Όλες οι απαντήσεις από τα παραπάνω services είναι σε JSON το οποίο μπορείτε να κάνετε parse ώστε να δημιουργήσετε ένα JavaScript αντικείμενο από την πλευρά της Javascript μέσω της εντολής `JSON.parse(json_str)`.

### **β) (10%) Αυτόματη συμπλήρωση του username βάσει φωτογραφίας**

Δημιουργήστε μία νέα html σελίδα για το sign-in ενός χρήστη με 2 textboxes, ένα για το username και ένα για το password. Επιπλέον εισάγετε και ένα radiobutton το οποίο όταν ενεργοποιείται θα επιτρέπει την αυτόματη συμπλήρωση του username βάσει της εικόνας του χρήστη. Για να το επιτύχετε αυτό θα χρησιμοποιήσετε το service Search (<https://console.faceplusplus.com/documents/5681455>). Σε περίπτωση επιτυχούς αναγνώρισης θα υπάρχει ένα σύνολο από usernames με κάποιο βαθμό εμπιστοσύνης (confidence), οπότε χρησιμοποιώντας εκείνο με το μεγαλύτερο βαθμό μπορείτε πλέον να συμπληρώσετε αυτόματα το username.

Αν έχετε κάποιο φίλο/η που παίρνει επίσης το μάθημα και έχει ανεβάσει μια εικόνα με το αντίστοιχο username μπορείτε να δείτε κατά πόσο θα τον αναγνωρίσει!

### ΠΡΟΣΟΧΗ!!!

- α) Το `faceplusplus` έχει πληρωθεί για ένα συγκεκριμένο αριθμό από requests οπότε αποφύγετε την άσκοπη χρήση του.
- β) Αν είστε ευαίσθητος/η σε θέματα ιδιωτικότητας, μπορείτε αντί να ανεβάζετε στο συγκεκριμένο service μία φωτογραφία σας, μπορείτε να χρησιμοποιήσετε τη φωτογραφία κάποιου ηθοποιού/τραγουδιστή (που καλό είναι να την έχετε τυπωμένη ώστε να τη χρησιμοποιήσετε και στη εξέταση!). Φροντίστε η φωτογραφία να είναι καθαρή και να φαίνεται καθαρά το πρόσωπο. Το συγκεκριμένο service αναφέρει ότι σώζει τις φωτογραφίες σε ασφαλές περιβάλλον και ότι δεν δίνει πρόσβαση σε τρίτους, αλλά ότι μπορεί να χρησιμοποιήσει τις φωτογραφίες για δικούς του σκοπούς.
- γ) Αν δεν διαθέτετε υπολογιστή με κάμερα μπορείτε απλά να τροποποιήσετε την άσκηση ώστε να κάνετε upload κάποια φωτογραφία από το filesystem ή από τον ιστό (Δείτε τις περιγραφές των services)!
- δ) Ο κώδικας που σας δόθηκε **παίξει μόνο σε chrome**. Επίσης λόγω του ότι το συγκεκριμένο αρχείο είναι στο file system και όχι σε κάποιο server ο chrome θα πρέπει να τρέξει με την παράμετρο `-allow-file-access-from-files` για να επιτραπεί η χρήση του stream με τη πληροφορία από την camera.
- ε) Χρησιμοποιείτε τα εργαλεία του browser για να βλέπετε τα requests/responses και τη console για τυχόν λάθη στη JS αλλά και μηνύματα που μπορείτε να γράφετε εσείς μέσω του `console.log`

## Άσκηση 2. `ourMap` Function [15%]

Στον παρακάτω κώδικα δηλώνουμε μία function με το όνομα **`ourMap`** η οποία δέχεται σαν παράμετρο ένα array με όνομα **`original`** και επιστρέφει μία function (ας την ονομάσουμε **`mapWithCallback`**) η οποία μπορεί να χρησιμοποιηθεί για να κάνουμε `map` μία συνάρτηση στα `elements` αυτού του array (δείτε τις διαλέξεις σχετικά με τις functions), δημιουργώντας ένα νέο array με το όνομα **`original`** και η οποία στη συνέχεια μπορεί να ξαναχρησιμοποιηθεί για να κάνει `map` μία άλλη συνάρτηση πάνω στα νέα στοιχεία του **`current`** array, κ.ο.κ. Άρα δηλαδή η **`mapWithCallback`** θα πρέπει να κάνει `track` ένα array που ονομάζεται **`current`** και το οποίο αρχικά ορίζεται να είναι ίδιο με **`original`** αλλά και το **`original`** array. Η **`mapWithCallback`** παίρνει 2 παραμέτρους:

- **`fn`** – Αν είναι function, τότε αυτή καλείται για κάθε στοιχείο του **`current`** array και το **`current`** array γίνεται `update` βάσει του αποτελέσματος της function. Αν η **`fn`** δεν είναι function τότε η επιστρεφόμενη τιμή της **`mapWithCallback`** θα είναι η τιμή του **`current`** array χωρίς να έχει γίνει κάποιο `map`.
- **`callback`** – Μία function που θα κληθεί όταν έχει γίνει το `map` πάνω στα στοιχεία του **`original`** array, το οποίο παίρνει σαν όρισμα. Το **`this`** μέσα στη callback θα πρέπει να κάνει reference στο **`current`** array. Αν η callback δεν είναι κάποια function, αγνοείται. Η return τιμή της callback δεν χρησιμοποιείται.

Η **`mapWithCallback`** επιστρέφει τον εαυτό της, ώστε να υποστηρίξουμε ένα μία αλυσίδα από ίδιες κλήσεις, εκτός και αν το `fn` argument δεν έχει οριστεί οπότε και επιστρέφει το **`current`** array. Θα πρέπει να είναι δυνατό να έχουμε πολλαπλές **`mapWithCallback`** functions την ίδια στιγμή.

Συμπληρώστε τις γραμμές που λείπουν στις γραμμές 1-16 με τον κατάλληλο κώδικα ώστε ο κώδικας που ακολουθεί να τρέχει σωστά. Δίνονται βοηθητικά σχόλια. Για κάθε γραμμή που συμπληρώνετε θα γίνεται και ένα commit όπου θα εξηγείτε το λόγο που τη συμπληρώσατε με αυτό τον τρόπο. Προσοχή, θα χρειαστεί να γίνει ένα κοινό commit για τις γραμμές 7,8 και άλλο ένα για τις γραμμές 12,13.

Επιπλέον σας δίνεται το test αρχείο **`A2test.js`** το οποίο έχει ενδεικτικές χρήσεις της **`ourMap`** και του τι αποτελέσματα θα πρέπει να επιστρέφει.

Προτείνεται η χρήση του online REPL για JavaScript `repl.it` (<https://repl.it>)

Προσπαθήστε να κατανοήσετε τα παραδείγματα που σας δίνονται στο test γιατί πάνω σε αυτά θα εξεταστείτε.

```
1 function ourMap(array) {  
2     // What do we have to hold/track???  
3     TO COMPLETE!
```

```

4      TO COMPLETE!
5      return (function mapWithCallback(fn, callback) {
6          // If fn is not a function return current Array
7          TO COMPLETE TOGETHER!
8          TO COMPLETE TOGETHER!
9          // map fn to array and save result to current array
10         TO COMPLETE!
11         // If callback is a function, execute callback
12         TO COMPLETE TOGETHER!
13         TO COMPLETE TOGETHER!
14         TO COMPLETE! // We have to return something!
15     })
16 }

```

Το υπόλοιπο 15% του βαθμού θα κατανεμηθεί βάσει των παρακάτω 3 κριτηρίων:

- **lint, code quality – 5%:** θα κρίνεται από το αν η σελίδα σας δεν εμφανίζει λάθη/warnings στον jslint/jshint, καθώς και στη γενική ποιότητα του κώδικά σας
- **ελκυστικότητα εμφάνισης σελίδων (στυλιστική συνέπεια) – 5%**
- **git – 5%:** θα κρίνεται η σωστή χρήση του git (π.χ. μικρά commits που να περιγράφουν με σαφήνεια πως κάνατε την άσκηση, κατανοητή περιγραφή, καθαρό ιστορικό, κτλ.)

Σημειώσεις:

Μη ξεχνάτε τη χρήση του "use strict"; για την JavaScript.

Το parse ενός JSON string γίνεται με χρήση της `JSON.parse(str)` που επιστρέφει το javascript object που αντιστοιχεί στο str, δεδομένου ότι το str είναι μια σωστή αναπαράσταση JSON .

Μπορείτε να χρησιμοποιήσετε κάποιον online linter όπως ο jslint <http://www.jshint.com/> ή ο jshint <http://jshint.com/> για να βελτιώσετε την ποιότητα του js κώδικά σας.

### Τρόπος Παράδοσης

Οι ασκήσεις θα παραδίδονται μόνο μέσω git, σύμφωνα με τις οδηγίες που σας έχουν δοθεί. Συγκεκριμένα στο repository σας στο bitbucket το οποίο θα πρέπει να έχει γίνει ήδη share στο hy359, στο folder a2 θα πρέπει να υπάρχουν δύο subfolders, **la** και **ourMap**, όπου θα περιέχεται ο κώδικας για κάθε άσκηση. **Θα πρέπει να φροντίσετε ότι όλα όσα έχετε κάνει έχουν γίνει σωστά commit και βρίσκονται online στο bitbucket. Αφού όλα είναι ok θα πρέπει να κάνετε tag την άσκηση σας "git tag a2" και να ανεβάσετε το tag και στο bitbucket μέσω της εντολής "git push --tags".**

**Προγραμματίστε καλά το χρόνο σας και αποφύγετε να ασχοληθείτε με την εργασία τελευταία στιγμή! Θα υπάρχουν αντίστοιχες ώρες γραφείου και τις 2 Πέμπτες!**

**Καλό είναι σαν τελευταίο έλεγχο πριν σταματήσετε να ασχολείστε με την εργασία να κάνετε clone το repository σας κάπου τοπικά και να ελέγξετε αν υπάρχει κάποιο πρόβλημα. Με αυτόν τον τρόπο άλλωστε θα βαθμολογηθείτε/εξεταστείτε.**

Στις 23:59 της 12/11 θα γίνει αυτόματο pull από όλα τα repositories που έχουν γίνει share στο hy359 και βάσει αυτών θα βαθμολογηθείτε. Εκπρόθεσμες ασκήσεις **δεν θα γίνονται δεκτές.**

### Αντιγραφή

Σε περίπτωση αντιγραφής θα μηδενίζονται άμεσα οι εργασίες όλων των εμπλεκόμενων.

### Και λίγα λόγια για το μεράκι....

Θέσετε ως στόχο να υλοποιήσετε τις εργασίες σας με μεράκι και δημιουργικότητα και να αναδείξετε με όμορφο τρόπο το χρόνο που αφιερώσατε! Μην αφήνετε τα πράγματα στην τύχη και δώστε κάτι από τον εαυτό σας και το χαρακτήρα σας!

Καλή εργασία