Εργασία 2026-27

Ανάπτυξη Κινητών Εφαρμογών

Σκοπός

Σκοπός της εργασίας είναι να δημιουργήσετε μια cross-platform εφαρμογή για κινητές συσκευές που βρίσκει σημεία ενδιαφέροντος (Points of Interest, POIs) γύρω από τον χρήστη και τα προβάλλει με περιεχόμενο από τα ανοικτά δεδομένα της Wikimedia (Wikidata, Wikipedia, Wikimedia Commons).

Σε αντίθεση με προηγούμενες χρονιές, η εφαρμογή **δεν διαθέτει δικό της backend**. Όλα τα δεδομένα ανακτώνται απευθείας από τη συσκευή μέσω των δημόσιων APIs της Wikimedia και αποθηκεύονται προσωρινά (cache) στη συσκευή. Οι χρήστες συνδέονται με τον λογαριασμό τους στη Wikimedia μέσω OAuth 2.0.

Η εφαρμογή περιλαμβάνει στοιχεία gamification: ο χρήστης «συλλέγει» POIs επισκεπτόμενός τα, απαντά σε κουίζ που παράγονται αυτόματα από τα δεδομένα του Wikidata και, προαιρετικά, συνεισφέρει στα έργα της Wikimedia συμπληρώνοντας ό,τι λείπει.

Διαδικαστικά Εργασίας

Η εργασία είναι **υποχρεωτική**, καλύπτει το **50%** της τελικής βαθμολογίας και είναι **ομαδική (έως 2 ατόμων)**. Ατομική εργασία μπορούν να υποβάλλουν μόνο φοιτητές πέραν του 4ου έτους (επί πτυχίω). Ο κώδικας φιλοξενείται σε **private repo στο GitHub** και στις ομαδικές εργασίες το πλήθος των commits ανά μέλος είναι κριτήριο συμβολής.

Η εφαρμογή πρέπει να εκτελείται μέσω **Expo Go** σε συσκευές Android και iOS και να λειτουργεί πλήρως κατά την εξέταση. Δεν απαιτείται RESTful API, Docker ή Swagger.

Κατά την εξέταση θα κληθείτε να επιδείξετε την εφαρμογή, να εξηγήσετε πώς υλοποιήσατε επιλεγμένες λειτουργίες και να υλοποιήσετε σε σύντομο χρόνο μια μικρή τροποποίηση (π.χ. «προσθέστε νέα κατηγορία» ή «κάντε ερώτηση κουίζ από το P84»), εν είδει προφορικής εξέτασης.

Υποστηρικτικά εργαστήρια

Κάθε τμήμα της εργασίας στηρίζεται σε εργαστήριο του μαθήματος. Μπορείτε (και πρέπει) να ξεκινήσετε από τον κώδικα των εργαστηρίων.

Εργαστήριο	Τι καλύπτει για την εργασία
Lab 04	Components, λίστες, `fetch`, loading / error states
Lab 05	React Navigation (stack, tabs), Context & `useReducer`
Lab 06	Θέση χρήστη, χάρτης, pins, απόσταση, check-in ≤100 m, προσομοίωση θέσης
Lab 07	Ερωτήματα A & B (SPARQL), `wbgetentities`, Wikipedia, Commons, User-Agent, HTTP 429, cache, offline
Lab 08	Σύνδεση με Wikimedia (OAuth 2.0 + PKCE), ασφαλής αποθήκευση & ανανέωση tokens, συγχρονισμός προόδου

Αντικείμενο Εργασίας

Καλείστε να υλοποιήσετε μια εφαρμογή σε **React Native (Expo)** που επικοινωνεί απευθείας με τα APIs της Wikimedia. Οι ελάχιστες τεχνικές απαιτήσεις είναι οι εξής.

Αυθεντικοποίηση με Wikimedia OAuth 2.0

Ο χρήστης συνδέεται με τον λογαριασμό του στη Wikimedia (**Authorization Code flow με PKCE, χωρίς client secret**).

- Κάθε ομάδα καταχωρεί **δική της** εφαρμογή (OAuth 2.0 consumer) στο meta.wikimedia.org: **μη εμπιστευτική** (Client is confidential: όχι), **όχι** owner-only, με **ελάχιστα** δικαιώματα. Οδηγίες: Εργαστήριο 8, «Η Φόρμα του Consumer, Πεδίο προς Πεδίο».
- Ως callback URL δηλώνεται η **σελίδα-γέφυρα** του μαθήματος, `https://riggas-ionio.github.io/mobile-apps/oauth/`, ώστε η σύνδεση να λειτουργεί μέσα από το Expo Go.
- Endpoints:
  - Authorize: `https://meta.wikimedia.org/w/rest.php/oauth2/authorize`
  - Token: `https://meta.wikimedia.org/w/rest.php/oauth2/access_token`
  - Στοιχεία χρήστη: `https://meta.wikimedia.org/w/rest.php/oauth2/resource/profile`
- Τα tokens αποθηκεύονται με ασφάλεια (`expo-secure-store`) και ανανεώνονται με το refresh token όταν λήγουν.
- Η περιήγηση στα POIs επιτρέπεται και σε **επισκέπτες**. Η σύνδεση απαιτείται για το προφίλ, τον συγχρονισμό προόδου και τις αποστολές συνεισφοράς.

Θέση χρήστη

Η εφαρμογή ζητά άδεια θέσης (`expo-location`). Αν ο χρήστης δεν τη δώσει, μπορεί να επιλέξει σημείο στον χάρτη (προεπιλογή: κέντρο της Κέρκυρας). Η εφαρμογή δείχνει **πάντα** στην οθόνη αν η θέση είναι από GPS ή επιλεγμένη / προσομοιωμένη.

Cache και σεβασμός στις υποδομές

- Κάθε αίτημα (και κάθε εικόνα) στέλνει header **`User-Agent`** (και `Api-User-Agent`) που αναγνωρίζει την εφαρμογή και την ομάδα, π.χ. `IonioPOI-team07/1.0 (https://github.com/…)`. Χωρίς αυτό η Wikimedia απαντά HTTP 403 (π.χ. στις εικόνες στο Android).
- Τα αποτελέσματα αποθηκεύονται τοπικά (`AsyncStorage` ή `expo-sqlite`) με κλειδί «κελί θέσης + ακτίνα + κατηγορία» (lat/lon στρογγυλεμένα στα 2 δεκαδικά) και **TTL 24 ωρών**.
- Δεν επιτρέπονται επαναλαμβανόμενα ερωτήματα σε κάθε μετακίνηση του χάρτη ή της θέσης: νέο ερώτημα γίνεται μόνο όταν αλλάζει το κελί.
- Η εφαρμογή εμφανίζει τα αποθηκευμένα POIs και **χωρίς σύνδεση** (offline), με σχετική ένδειξη.

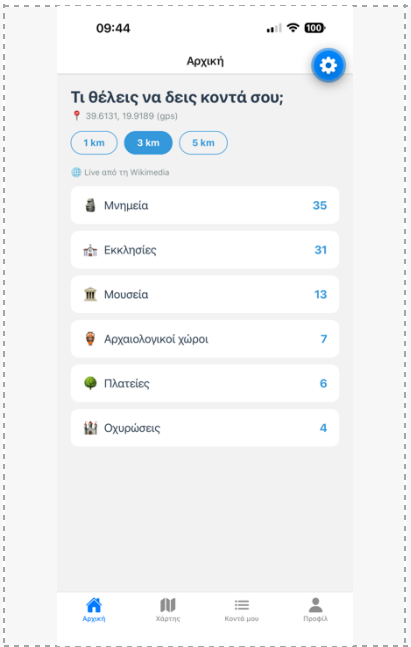
Απαιτούμενες οθόνες

Η εφαρμογή πρέπει να παρέχει κατ' ελάχιστο τις ακόλουθες έξι οθόνες:

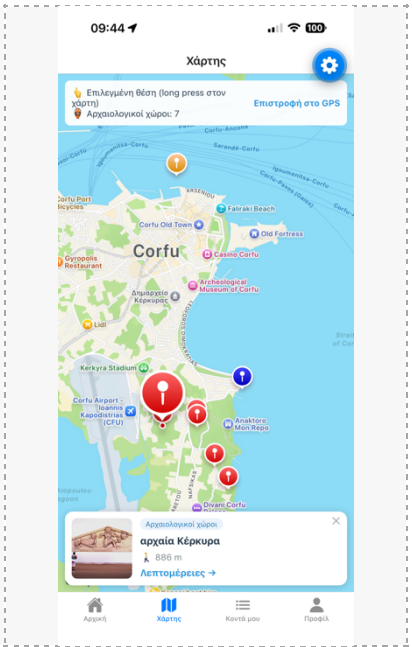
- **Σύνδεση και ομάδα.** Κουμπί «Σύνδεση με Wikimedia», επιλογή «Συνέχεια ως επισκέπτης» και, κάτω από αυτά, τα ονοματεπώνυμα και οι ΑΜ των μελών της ομάδας.
- **Αρχική: «Τι θέλεις να δεις κοντά σου;»** Η θέση του χρήστη, επιλογή ακτίνας (1 / 3 / 5 km) και 4–6 κουμπιά κατηγοριών με το πλήθος των POIs τους. Κουμπιά εμφανίζονται μόνο για κατηγορίες με αποτελέσματα (Ερώτημα Α).
- **Χάρτης κατηγορίας.** Χάρτης πλήρους οθόνης με τα POIs (Ερώτημα Β) και λίστα ταξινομημένη κατά απόσταση. Τα pins έχουν διαφορετικό χρώμα για συλλεγμένα, μη συλλεγμένα και «χρειάζονται φωτογραφία». Το tap σε pin δείχνει προεπισκόπηση (tooltip ή κάρτα) με τίτλο, μικρή φωτογραφία και απόσταση.
- **Πλήρης προβολή POI** (`wbgetentities`), με:
  - slider ≥3 φωτογραφιών, η πρώτη από το P18 (οι υπόλοιπες από την κατηγορία Commons, P373), όπου υπάρχουν
  - την κατηγορία ως tag, τίτλο, σύντομη περιγραφή και συντεταγμένες από το Wikidata
  - εκτενέστερο κείμενο από τη Wikipedia (REST summary) και σύνδεσμο στο λήμμα
  - κουμπί «Είμαι εδώ» (check-in), ενεργό μόνο σε απόσταση ≤100 m
- **Κουίζ.** Μία έως τρεις ερωτήσεις πολλαπλής επιλογής μετά το check-in, με άμεση ανατροφοδότηση και πόντους.
- **Προφίλ.** Συλλογές ανά κατηγορία, badges, συνολικοί πόντοι και ενεργές αποστολές.



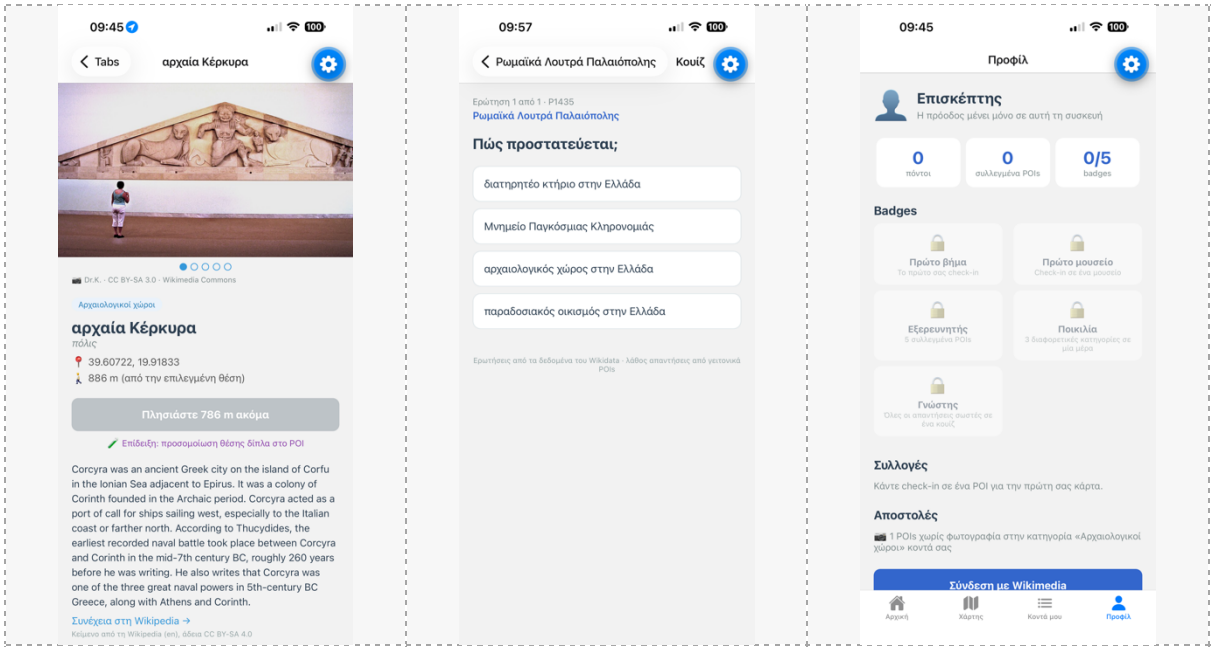
Σύνδεση και ομάδα
OAuth 2.0 / επισκέπτης



Αρχική
Ερώτημα Α



Χάρτης κατηγορίας
Ερώτημα Β



Πλήρης προβολή POI
wbgetentities, Wikipedia, Commons

Κουίζ
P571 / P84 / P138 / P1435

Προφίλ
πόντοι, badges, συλλογές

Χάρτης

Χρησιμοποιήστε το `react-native-maps` (λειτουργεί στο Expo Go χωρίς API key) ή, εναλλακτικά, Leaflet + OpenStreetMap σε `WebView` (Lab 06). Αν φτιάξετε δικό σας APK με `react-native-maps`, θα χρειαστείτε δικό σας Google Maps API key — η επίδειξη όμως γίνεται στο Expo Go.

Κατηγορίες POIs και ερωτήματα Wikidata

Από πού έρχονται τα δεδομένα: SPARQL ή HTTP API;

Και τα δύο, για διαφορετική δουλειά. Και τα δύο είναι κλήσεις HTTP με `fetch` που επιστρέφουν JSON. Τα ερωτήματα **SPARQL** στο Wikidata Query Service (`https://query.wikidata.org/sparql?query=…&format=json`) χρησιμοποιούνται για την **αναζήτηση**: ποια POIs υπάρχουν γύρω από τη θέση, ανά κατηγορία και μαζί με τις υποκλάσεις. Τα **HTTP APIs** της Wikimedia χρησιμοποιούνται για τις **λεπτομέρειες** ενός POI. Το SPARQL είναι πιο αργό και έχει αυστηρότερα όρια (HTTP 429), γι' αυτό καλείται μία φορά ανά κελί θέσης και κατηγορία και το αποτέλεσμα αποθηκεύεται στην cache.

Τι χρειάζεται η εφαρμογή	Από πού	Οθόνη
POIs γύρω μου ανά κατηγορία και πλήθος ανά κατηγορία	**SPARQL** (Ερωτήματα A και B)	Αρχική, Χάρτης
Όλα τα στοιχεία ενός POI (ετικέτες, statements, sitelinks)	Wikidata Action API, `wbgetentities`	POI
Κείμενο λήμματος	Wikipedia REST, `page/summary`	POI
Φωτογραφίες, δημιουργός και άδεια	Commons: `Special:FilePath`, `prop=imageinfo`	POI, προεπισκόπηση

Τι χρειάζεται η εφαρμογή	Από πού	Οθόνη
Λάθος απαντήσεις κουίζ από γειτονικά POIs	**SPARQL** (όπως το παράδειγμα με το P571)	Κουίζ
Σύνδεση και στοιχεία χρήστη	OAuth REST στο meta.wikimedia.org	Σύνδεση, Προφίλ

Η αναζήτηση μπορεί να γίνει και χωρίς SPARQL, με το Action API (βλ. το τέλος αυτής της ενότητας).

Τα POIs δεν είναι σταθερά, αλλά ανακτώνται δυναμικά γύρω από τη θέση του χρήστη. Κάθε ομάδα διαλέγει **4–6 κατηγορίες**. Ενδεικτικές κλάσεις (με το πλήθος σε ακτίνα 3 km από το κέντρο της Κέρκυρας, Νοέμβριος 2026):

Κουμπί	Κλάση Wikidata	Κέρκυρα, 3 km
Μνημεία	`wd:Q4989906` (monument)	35
Εκκλησίες	`wd:Q16970` (church building)	31
Μουσεία	`wd:Q33506` (museum)	13
Αρχαιολογικοί χώροι	`wd:Q839954` (archaeological site)	7
Πλατείες	`wd:Q174782` (square)	6
Οχυρώσεις	`wd:Q57821` (fortification)	4

Το `wdt:P31/wdt:P279*` πιάνει και τις υποκλάσεις (π.χ. το αρχαιολογικό μουσείο μετρά στα μουσεία). Η κλάση «κάστρο» (`wd:Q23413`) δίνει ο στην Κέρκυρα — το Παλαιό Φρούριο είναι οχύρωση. Στο WKT η σειρά είναι `Point(lon lat)` και η ακτίνα δίνεται σε km.

Ερώτημα A: πλήθος POIs ανά κατηγορία (αρχική οθόνη)

```
SELECT ?cat (COUNT(DISTINCT ?item) AS ?n) WHERE {
  VALUES ?cat { wd:Q33506 wd:Q16970 wd:Q4989906 wd:Q174782 wd:Q57821 wd:Q839954 }
  SERVICE wikibase:around {
    ?item wdt:P625 ?coord .
    bd:serviceParam wikibase:center "Point(19.92 39.62)"^^geo:wktLiteral ;
                    wikibase:radius "3" .
  }
  ?item wdt:P31/wdt:P279* ?cat .
}
GROUP BY ?cat
ORDER BY DESC(?n)
```

Ερώτημα B: POIs μιας κατηγορίας (χάρτης)

```
SELECT DISTINCT ?item ?label ?desc ?coord ?dist ?image ?elwiki ?enwiki WHERE {
  SERVICE wikibase:around {
    ?item wdt:P625 ?coord .
    bd:serviceParam wikibase:center "Point(19.92 39.62)"^^geo:wktLiteral ;
                    wikibase:radius "3" ;
                    wikibase:distance ?dist .
  }
  ?item wdt:P31/wdt:P279* wd:Q33506 .
  OPTIONAL { ?item wdt:P18 ?image }
  OPTIONAL { ?elwiki schema:about ?item ; schema:isPartOf <https://el.wikipedia.org/> }
  OPTIONAL { ?enwiki schema:about ?item ; schema:isPartOf <https://en.wikipedia.org/> }
  SERVICE wikibase:label {
    bd:serviceParam wikibase:language "el,en,mul" .
    ?item rdfs:label ?label ; schema:description ?desc .
  }
}
ORDER BY ?dist
LIMIT 50
```

Ένα αντικείμενο με δύο φωτογραφίες ή δύο συντεταγμένες επιστρέφει **δύο γραμμές** — κρατήστε ένα POI ανά Q-ID.

Αλλαγή backend του Wikidata Query Service

Το Wikidata Query Service μεταφέρεται σε νέα μηχανή (QLever), η οποία δεν υποστηρίζει τα `wikibase:around` και `wikibase:label`. Η εφαρμογή πρέπει να κρατά **endpoint και μορφή ερωτήματος σε ένα σημείο** του κώδικα (π.χ. `api/wikidata.js`), ώστε η αλλαγή να γίνεται χωρίς να αγγίξετε τις οθόνες. Η μορφή v2 του Ερωτήματος Β (με `geof:distance` και `rdfs:label` αντί για τις υπηρεσίες `wikibase:*`) δίνεται στην ηλεκτρονική εκδοχή της εκφώνησης.

Λεπτομέρειες POI, κείμενο και εικόνες

- Δεδομένα αντικειμένου: `https://www.wikidata.org/w/api.php?action=wbgetentities&ids=${id}&props=labels|descriptions|claims|sitelinks&languages=el|en&format=json`
- Σύνοψη λήμματος: `https://el.wikipedia.org/api/rest_v1/page/summary/${title}`
- Εικόνα από το P18: `https://commons.wikimedia.org/wiki/Special:FilePath/${photoname}?width=1000`. Αντικαθιστά το URL του τύπου `/thumb/4/4a/…` της προηγούμενης χρονιάς, γιατί η διαδρομή `4/4a` αλλάζει ανά αρχείο (MD5 του ονόματος).
- Δημιουργός & άδεια φωτογραφιών: `https://commons.wikimedia.org/w/api.php?action=query&prop=imageinfo&iiprop=url|extmetadata&titles=File:…`

Εναλλακτικά, οι ομάδες μπορούν να χρησιμοποιήσουν το Action API χωρίς SPARQL: `list=geosearch` στο `wikidata.org` (`gscoord=39.62|19.92&gsradius=3000&gslimit=100`), έπειτα `wbgetentities` σε ομάδες των 50 ids και φιλτράρισμα βάσει P31 στον κώδικα. Σε αυτή την περίπτωση δεν πιάνονται οι υποκλάσεις, οπότε κάθε κατηγορία χρειάζεται λίστα Q-IDs.

Gamification

Τα Επίπεδα 1 και 2 είναι υποχρεωτικά. Ο συγχρονισμός προόδου και το Επίπεδο 3 είναι προαιρετικά και δίνουν bonus.

Επίπεδο	Τι κάνει ο χρήστης	Πόντοι (ενδεικτικά)	Πού αποθηκεύονται
1. Εξερευνητής	Check-in σε POI (≤100 m)	10 ανά POI	Τοπικά
2. Κουίζ	Σωστή απάντηση μετά το check-in	5 ανά ερώτηση	Τοπικά
Bonus: Συγχρονισμός	Η πρόοδος στον λογαριασμό του χρήστη	—	`action=options` (`userjs-…`)
3. Αποστολές συνεισφοράς (bonus)	Φωτογραφία POI χωρίς P18 και ανέβασμα	30 ανά αρχείο	Στα ίδια τα wikis (επαληθεύσιμα)

Επίπεδο 1: Εξερευνητής

- Το check-in επιτρέπεται μόνο όταν η απόσταση από το P625 είναι ≤100 m (τύπος haversine στον κώδικα).

- Το συλλεγμένο POI γίνεται «κάρτα» με τη φωτογραφία του P18, και οι κάρτες ομαδοποιούνται σε συλλογές ανά κατηγορία.
- Κάθε ομάδα ορίζει τουλάχιστον **3 badges** (π.χ. «Πρώτο μουσείο», «5 πλατείες», «3 κατηγορίες σε μία μέρα»).
- Η πρόοδος αποθηκεύεται στη συσκευή και **δεν χάνεται** όταν κλείνει η εφαρμογή.

Bonus: Συγχρονισμός προόδου

Η πρόοδος συγχρονίζεται στον λογαριασμό του χρήστη με `action=options` (κλειδί `userjs-<εφαρμογή>`, μικρό JSON), ώστε να διατηρείται σε άλλη συσκευή χωρίς backend (Εργαστήριο 8). Απαιτεί το δικαίωμα `editmyoptions`, άρα ο consumer χρειάζεται έγκριση από τους OAuth admins — μέχρι τότε λειτουργεί για τον λογαριασμό που τον δημιούργησε, κάτι που αρκεί για την επίδειξη.

Επίπεδο 2: Κουίζ από τα linked data

Μετά το check-in η εφαρμογή παράγει αυτόματα ερώτηση από ένα statement του αντικειμένου. Η σωστή απάντηση είναι η τιμή του POI και οι λάθος απαντήσεις προέρχονται από γειτονικά POIs. Απαιτούνται τουλάχιστον **2 τύποι ερώτησης** από τις εξής ιδιότητες:

- **P571** (έτος ίδρυσης / κατασκευής): «Πότε χτίστηκε;»
- **P84** (αρχιτέκτονας): «Ποιος το σχεδίασε;»
- **P138** (πήρε το όνομά του από): «Ποιον τιμά το όνομα;»
- **P1435** (καθεστώς προστασίας): «Πώς προστατεύεται;»

Παράδειγμα ερωτήματος για ερωτήσεις έτους:

```
SELECT ?item ?label ?year WHERE {
  SERVICE wikibase:around {
    ?item wdt:P625 ?c .
    bd:serviceParam wikibase:center "Point(19.92 39.62)"^^geo:wktLiteral ;
                    wikibase:radius "3" .
  }
  ?item wdt:P571 ?inception .
  BIND(YEAR(?inception) AS ?year)
  SERVICE wikibase:label { bd:serviceParam wikibase:language "el,en,mul" .
                           ?item rdfs:label ?label }
}
LIMIT 30
```

Αν το POI δεν έχει καμία από τις ιδιότητες, η εφαρμογή δεν δείχνει κουίζ και ενημερώνει τον χρήστη ότι λείπουν δεδομένα στο Wikidata.

Επίπεδο 3: Αποστολές συνεισφοράς (προαιρετικό, bonus)

- Η εφαρμογή δείχνει κοντινά αντικείμενα χωρίς φωτογραφία: στο Ερώτημα Β, το `OPTIONAL { ?item wdt:P18 ?image }` γίνεται `FILTER NOT EXISTS { ?item wdt:P18 [] }`.
- Ο χρήστης τραβάει φωτογραφία επιτόπου και την ανεβάζει με το OAuth token του (`action=upload`), με άδεια CC BY-SA 4.0 και λεζάντα που συμπληρώνει ο χρήστης.
- Κατά την ανάπτυξη και την επίδειξη, τα ανεβάσματα και οι επεξεργασίες γίνονται **μόνο στα test wikis** (test.wikipedia.org, test.wikidata.org).
- Οι πόντοι του Επιπέδου 3 υπολογίζονται από τις πραγματικές συνεισφορές (`list=allimages&aiuser=…` ή `list=usercontribs`), δεν αποθηκεύονται στη συσκευή.

Καλές πρακτικές προς την κοινότητα Wikimedia

Η εφαρμογή σας χρησιμοποιεί κοινόχρηστες υποδομές και περιεχόμενο που δημιουργούν εθελοντές. Οι παρακάτω κανόνες είναι **υποχρεωτικοί** και ελέγχονται στην αξιολόγηση.

- **Σεβασμός στις υποδομές**: User-Agent που αναγνωρίζει την εφαρμογή, cache, όχι παράλληλα ερωτήματα στο WDQS, σωστός χειρισμός του HTTP 429 (αναμονή όσο λέει το `Retry-After`).
- **Αναφορά πηγών και αδειών**: κάθε φωτογραφία δείχνει δημιουργό και άδεια (από `prop=imageinfo&iiprop=extmetadata` στο Commons) και το κείμενο από τη Wikipedia συνοδεύεται από σύνδεσμο και άδεια CC BY-SA.
- **Καμία επεξεργασία στα πραγματικά wikis χωρίς ρητή ενέργεια του χρήστη**: το Επίπεδο 3 ζητά επιβεβαίωση πριν από κάθε ανέβασμα, και κατά την ανάπτυξη χρησιμοποιούνται μόνο τα test wikis.
- **Δεν αλλοιώνετε τα δεδομένα για να «περάσει» η εφαρμογή**: αν ένα POI έχει λάθος ή ελλιπή δεδομένα, η εφαρμογή το χειρίζεται αθόρυβα. Διορθώσεις στο Wikidata γίνονται μόνο αν είναι τεκμηριωμένες, με πηγή.
- **Ιδιωτικότητα**: η θέση του χρήστη δεν στέλνεται πουθενά αλλού εκτός από τα ερωτήματα προς τη Wikimedia (στρογγυλεμένη στα 2 δεκαδικά) και δεν αποθηκεύεται ιστορικό διαδρομών.
- **Μυστικά**: κανένα token, κωδικός ή client secret στον κώδικα ή στο GitHub.

Κριτήρια αξιολόγησης

Κριτήριο	Μονάδες
Αυθεντικοποίηση OAuth 2.0 PKCE, ασφαλής αποθήκευση και ανανέωση tokens	15
Θέση χρήστη και δυναμικά κουμπιά κατηγοριών (Ερώτημα Α)	10
Χάρτης, pins και προεπισκόπηση (Ερώτημα Β)	15
Πλήρης προβολή POI (slider, κείμενο, αναφορά πηγών)	15
Cache, offline λειτουργία, User-Agent και χειρισμός σφαλμάτων	10
Επίπεδο 1: check-in, συλλογές, badges	15
Επίπεδο 2: κουίζ από τουλάχιστον 2 ιδιότητες	10
Ποιότητα κώδικα, commits, README	10
Σύνολο	**100**
Bonus: συγχρονισμός προόδου με `action=options`	+5
Bonus: Επίπεδο 3 (αποστολές συνεισφοράς στα test wikis)	+10

Η προφορική εξέταση και η τροποποίηση κώδικα στην παρουσίαση λειτουργούν ως πολλαπλασιαστής ατομικής συμβολής.

Τελικά Παραδοτέα και Ημερομηνία

Η παρουσίαση γίνεται υποχρεωτικά δια ζώσης *την τελευταία εβδομάδα μαθημάτων* πριν την εξεταστική· μόνο για φοιτητές ≥4ου έτους η παρουσίαση γίνεται στην ημέρα και ώρα εξέτασης που

θα ανακοινωθεί στο πρόγραμμα εξεταστικής. Κατά την επίδειξη, το check-in μπορεί να γίνει με **προσομοιωμένη θέση**, εφόσον η εφαρμογή δείχνει στην οθόνη ότι η θέση είναι προσομοιωμένη.

Πριν την παρουσίαση, **οι φοιτητές όλων των ετών** υποβάλλουν στο opencourses:

1) Ένα **README.md** με τα ονοματεπώνυμα και τους ΑΜ των μελών, τις κατηγορίες που επέλεξαν με τα Q-IDs τους, το User-Agent της εφαρμογής, το client ID του OAuth consumer και τους κανόνες gamification (πόντοι, badges).

2) Μια **μαγνητοσκοπημένη επίδειξη** (screen capture με **voice over**) της κινητής εφαρμογής σας[1] ως σύνδεσμο MS Stream (μην την ανεβάσετε στο opencourses), με δικαίωμα αναπαραγωγής και λήψης για τους διδάσκοντες (riggas@ionio.gr, dimitrakopoulos@ionio.gr). Η επίδειξη καλύπτει σύνδεση, θέση, check-in (με πραγματική ή προσομοιωμένη θέση), κουίζ και offline λειτουργία.

3) **Όλο τον source κώδικα** (όχι το φάκελο node_modules) **σε zip**. Η εφαρμογή πρέπει να ξεκινά με **npm install** και **npx expo start**, χωρίς άλλες υπηρεσίες.

4) **Στιγμιότυπα οθόνης από τις δικές σας συσκευές** (PNG ή JPG, σε ξεχωριστό zip): ένα τουλάχιστον για καθεμία από τις 6 βασικές οθόνες (Σύνδεση, Αρχική, Χάρτης κατηγορίας με προεπισκόπηση, Πλήρης προβολή POI, Κουίζ, Προφίλ), με ονόματα όπως *01-login.png*, *02-home.png* κ.λπ. Τα στιγμιότυπα προέρχονται από την εφαρμογή **σε πραγματική λειτουργία σε κινητό** (όχι από τον browser, ούτε mockups του Figma), με πραγματικά δεδομένα από τη Wikimedia. Κάθε μέλος της ομάδας τραβά τουλάχιστον ένα από τη δική του συσκευή· αν η ομάδα έχει και iOS και Android, δείξτε και τα δύο. Προαιρετικά, προσθέστε ένα από την offline λειτουργία ή από ένα μήνυμα σφάλματος.

Σημείωση για την εμφάνιση: οι οθόνες που περιγράφονται στην εκφώνηση είναι ενδεικτικές ως προς τη λειτουργία, όχι ως προς το στυλ. Θα εκτιμηθεί ιδιαίτερα μια **δική σας αισθητική ταυτότητα**: παλέτα χρωμάτων, τυπογραφία, εικονίδια, λογότυπο, κάρτες και μικρο-κινήσεις, σχεδιασμένα από την ομάδα (π.χ. στο Figma, όπως στο Εργαστήριο 9). Εφαρμογές που μένουν στην προεπιλεγμένη εμφάνιση ενός εργαλείου AI ή του υλικού των εργαστηρίων δεν θα ξεχωρίσουν· θέλουμε να αναγνωρίζουμε την εφαρμογή σας από ένα στιγμιότυπο.

Όλες οι ομάδες υποβάλλουν πλήρως τα παραδοτέα τους πριν την παρουσίαση της τελευταίας εβδομάδας, ανεξάρτητα αν είναι μεγαλύτερου έτους και θα παρουσιάσουν εντός της εξεταστικής. Οι ακριβείς ημερομηνίες θα ανακοινωθούν μαζί με το πρόγραμμα της εξεταστικής.

-- Καλή επιτυχία --

[1] *Μπορείτε να κάνετε screen recording με χρήση όποιου εργαλείου προτιμάτε και μετά να γίνει upload στο MS Stream*